Journal of ELECTRICAL ENGINEERING, VOL 72(2021), NO2, 89–98

sciendo

PAPERS

A study of securing in-vehicle communication using IPSEC protocol

Jan Lastinec[1] **, Ladislav Hudec**[1]

Current vehicles are increasingly dependent on Electronic Control Units (ECUs) that control virtually every system of the vehicle. To enable advanced features automotive embedded systems are opening to external world, which raises security concerns. At the same time these innovative systems require more complex software and higher bandwidth for information exchange. Thanks to its bandwidth, payload size, and openness, Ethernet is a candidate technology for future in-vehicle architectures. This paper deals with design of a novel approach to secure In-vehicle Systems by taking advantage of Ethernet/IP technology and proven security mechanisms from TCP/IP model. Main goal is to design an efficient solution that meets requirements for latency without requiring high amounts of processing power and provides secure exchange of control messages. The work is mainly focused on the widespread Controller Area Network (CAN). The presented solution is based on encapsulation of CAN frames into UDP datagrams with added authenticity, integrity, and (if required) confidentiality of communication using IPsec protocol in transport mode. This creates a "secure tunnel across backbone Ethernet network in a vehicle. Next part of the paper presents extensive tests in simulation that are based on our previous experiments on hardware, in order to evaluate the characteristics of the designed security extension. The results indicate that using IPsec is a viable solution for securing in-vehicle communications.

K e y w o r d s: controller area network, automotive ethernet, TCP/IP, communication security, IPsec

1 Introduction

Modern vehicles utilize several tens of Electronic Control Units (ECUs) to implement advanced driver assistance features. The development of infotainment systems that provide Internet connectivity, synchronization with smartphones and other functions is proceeding at fast rate as well. However, the transition from closed systems to open architecture exposes vehicle infrastructure to different threats and attack vectors [1–3].

By design, controller area network [4] provides reliable and fault-tolerant communication. On the other hand, basic security properties such as authenticity, integrity, freshness of exchanged CAN frames, are not present. In-vehicle network infrastructure is commonly composed of several subnetworks interconnected by a CAN backbone. The connections are made through gateway ECUs. Access to the control CAN is only as secure, as are the border ECUs that connect it to the outside world or other subnetworks.

A study of cryptographic approaches to secure CAN bus communication [5] shows that it is possible to increase the security of CAN bus communication directly using software-based security mechanisms. However, the latency is increased by 60–70 % and it introduces significant negative consequences (payload limitations, need for transmitting multiple frames, *etc*). The processing overhead of software security mechanisms needs to be taken into account, especially for low-performance ECUs [6]. A hardware-based approach to security makes use of Hardware Security Modules (HSM) [7]. HSM safeguards cryptographic material storage and provides various levels of hardware acceleration for cryptographic algorithms. Even with increased cryptographic performance, it is still necessary to reserve payload space for Message Authentication Codes (MAC). The maximum latency varies in range from 4 to 9 ms based on the MAC length (32–512 bits).

With increasing demands for connectivity and bandwidth current centralized in-vehicle architecture is slowly reaching its limits. Thanks to its bandwidth and openness, Ethernet is considered a suitable candidate technology for backbone network in future hierarchical in-vehicle networks. Works dealing with automotive Ethernet are mostly oriented on evaluating its characteristics and suitability for automotive domain [8, 9]. The transition to Ethernet is expected to be of incremental fashion. Therefore, possibilities of transforming existing CAN communication into Ethernet/IP are researched as well. [10] evaluated different strategies of transforming CAN frames to Ethernet/IP packets on a CAN/Ethernet gateway. The experiments were implemented on FPGA prototypes and the end-to-end 0delay between two CAN networks varied from 500 μs to 4500 μs based on the used gateway strategy.

To summarise, existing automotive bus systems provide very limited space for incorporating security mechanisms and performance evaluations of Ethernet in vehicles aim at safety and dependability but not security. Ethernet and especially TCP/IP model provide mature

Institute of Computer Engineering and Applied Informatics, FIIT STU, Bratislava 84216, Slovakia, jan.lastinec@stuba.sk

https://doi.org/10.2478/jee-2021-0012, Print (till 2015) ISSN 1335-3632, On-line ISSN 1339-309X

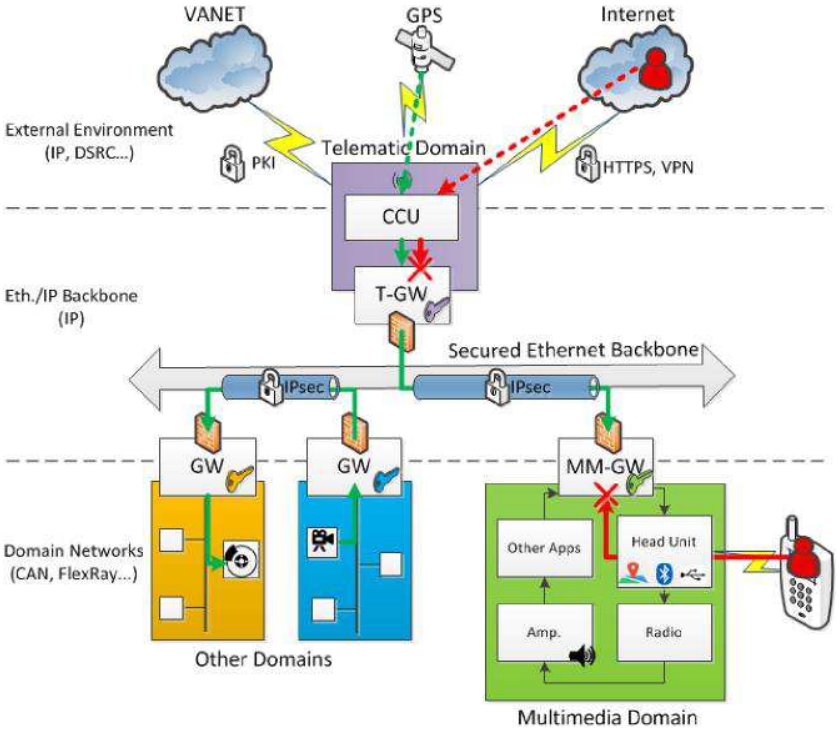


Fig. 1. Proposed security design. (CCU – communication control unit, GW – gateway, MM-GW – multimedia gateway, T-GW – telematics gateway)

security mechanisms that have potential to increase the security of automotive communication.

This paper is concerned with designing a security extension of automotive communication protocols leveraging Ethernet/IP and reusing proven TCP/IP security solution. We are focused on controller area network (CAN) which is currently the most widespread communication bus in automobiles. The rest of the paper is organised as follows. Next Section reviews assumptions and considerations. The proposal is described in Section 3 and evaluation of the presented solution is found in Section 4. The last Section contains discussion and conclusions.

2 Assumptions and considerations

From the security point of view, the most vulnerable part of CAN protocol is the frames being broadcasted to every node on the bus without any means to verify their origin. This means that one ECU could potentially impersonate another ECU. In order to guarantee the origin of received control data, and that the data have not been maliciously modified or replayed by a potential attacker, maintaining authenticity and integrity of the frames is a priority. Control data disclosure does not directly affect vehicle security so we consider confidentiality less important in this paper.

We assume that attacker does not have physical access to the vehicles CAN bus. In other words, the aim is to secure the on-board network against attacks from outside (Internet, malicious devices, *etc*). No additional restrictions are placed on the attacker.

A domain-based architecture of in-vehicle network with Ethernet/IP backbone [11] is considered in this paper. It is a hierarchical concept where respective subsystems are divided into several domains according to their functionality. The communication within domains is managed by so-called "domain controllers that are interconnected via Ethernet/IP backbone. Domain controllers are high-performance ECUs that provide access to/from the underlying networks. Because the domain controllers act as gateways for the domain networks, they are referred to as Domain Gateways, or simply Gateways ("GW) in this document. Due to integrating more advanced features in modern vehicles, the complexity of automotive E/E (electrical and electronic) systems increases. Currently widespread centralised architecture consists of mostly incompatible proprietary bus technologies and appears to be a limiting factor in manufacturers innovation efforts. Compared to the centralised approach, domain-based architecture is scalable and by utilizing an Ethernet/IP backbone it simplifies the exchange of messages between different functional domains. Furthermore, it can provide backwards-compatibility by allowing the sub-networks within a domain to use traditional bus technologies. For these reasons, domain-based architecture is considered a probable candidate for future in-vehicle network.

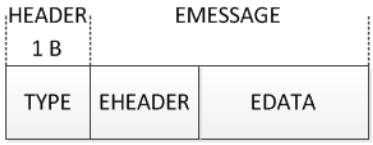


Fig. 2. Encapsulation protocol

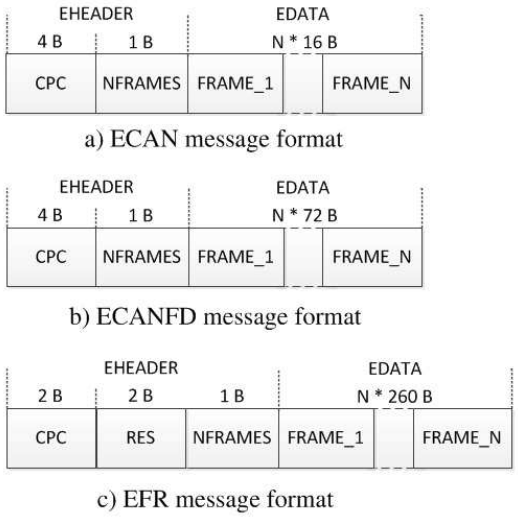


Fig. 3. EMESSAGE formats

Table 1. TYPE field definition

Byte	Bit	Field	Description
	[7:3]	RES	Reserved for future use
0	[2:0]	ENCP	Encapsulated protocol: 000 - ECAN (Controller Area Network) 001 - ECANFD (CAN Flexible Datarate) 010 - EFR (FlexRay)

Table 2. EFR - EDATA frame definition

Byte	Field	Description
[0:1]	ID	FlexRay Frame ID (11 bits) + type bits (5 bits).
[2]	LENGTH	Payload Length in 2-byte words (0 - 127).
[3]	CYCLE	FlexRay Cycle Count (0 - 63).
[4:5]	RES	Reserved for future use.
[6:259]	DATA	FlexRay Frame Payload (254 B).

In order to guarantee real-time properties of control communication we consider 10 ms hard deadline for delivering messages between nodes [9].

3 Security extension

Solution proposed in this work extends hierarchical concept of automotive network by adding security services to the communication occurring on the backbone layer. Main focus is put on increasing the security properties of CAN Bus but the design allows extension to other in-vehicle technologies as well. In order to keep the impact of security on the performance low, CPU-intensive security processing is offloaded from regular control units to powerful Domain Gateways, therefore the communication is secured on the Ethernet/IP backbone.

The proposed domain-based architecture with secured backbone is depicted in Fig. 1. It is divided into three layers: bottom layer represents existing domain networks that employ standard automotive protocols such as CAN, FlexRay, *etc.* Apart from adding the gateways (GW), it is not necessary to alter them in any way. The middle layer consists of Ethernet IP backbone that interconnects domain gateways. The communication is secured on this layer by guaranteeing authenticity and integrity using IPsec protocol and symmetric keys. Therefore, it is possible to verify the source GW of each message and prevent unwanted communication between domains. Depicted example shows allowing communication to Chassis domain (orange box) only from ADAS (Advanced Driver Assistance Systems) domain (blue box) and denying communication attempts from the Internet and Multimedia domain. The third layer shows possible integration of proposed architecture with environments external to vehicle, such as Internet or VANET where communication could be secured using HTTPS, VPN or Public Key Infrastructure (PKI). This layer is out of scope of this paper.

Gateway is an essential element of the security architecture. It performs the following functions:

- encapsulation/decapsulation according to proposed encapsulation protocol,
- traffic forwarding between domain network and Ethernet/IP backbone supporting multiple gateway strategies,
- security layer operations to maintain secure communication on the Ethernet backbone, and access control to its underlying domain.

3.1 Encapsulation protocol

A self-designed protocol created in this work allows automotive frames to be encapsulated into Ethernet/IP packets without losing any information. The aim is to provide a method of communicating between several domain networks through Ethernet network (*ie* tunnelling domain network traffic through UDP/TCP). Thanks to the encapsulation into transport layer segments, it is possible to engage security protocols from TCP/IP model to secure backbone messages. Main features of the protocol are:

- support for both UDP and TCP transport thanks to the fact that it operates on the application layer,
- extensible to support different automotive bus technologies (this work is oriented on CAN),
- support for N to 1 mapping of domain network frames into transport layer segments,
- message priority based on the encapsulated data (*eg* CAN ID).

All messages of the protocol share the same general format which is shown in Fig. 2. It consists of a HEADER with TYPE field and encapsulated message (EMESSAGE). TYPE specifies the encapsulated protocol as defined in Table 1. EMESSAGE contains its own header and data fields (EHEADER and EDATA) based on the protocol type specified in ENCP field. EMESSAGE definitions for encapsulated protocols are given in Fig. 3.

ECAN is EMESSAGE definition for encapsulating CAN frames. ECAN message format is depicted in Fig. 3(a) and it consists of: CPC (Content Priority Code) which defines the priority of message based on CAN identifiers of encapsulated frames in EDATA part. In case there are multiple EMESSAGEs to be processed this value can be used by receiving Gateway to quickly determine processing order. NFRAMES holds the number of frames in the payload. EDATA part consists of individual encapsulated frames (FRAME_1..N), stored in SocketCAN format [12]. The CPC value is calculated as the lowest ID of the encapsulated frames:

$$CPC = \min\big(id(Frame\_1), \ldots, id(FRAME\_N)\big)\,, \quad (1)$$

where $id(f)$ denotes CAN ID of the respective frame f. Therefore, in case of one-to-one mapping CPC value and CAN ID value of encapsulated frame are equal.

ECANFD is EMESSAGE definition for encapsulating CAN FD frames and its format is identical to CANFD, except the length of frames in EDATA part (Fig. 3(b)) that use SocketCAN format for CAN FD (72 bytes per frame).

EFR is EMESSAGE definition for encapsulating FlexRay frames and the format of EFR header is given in Fig. 3(c). NFRAMES field has identical meaning as in ECAN and ECANFD. CPC serves the same function but it operates with FlexRay Frame IDs instead of CAN IDs. RES is reserved for possible future usage. Encapsulated FlexRay frames in EDATA part have format as described by Table 2.

3.2 Gateway strategies

Essentially, there are two possible ways of embedding CAN frames into Ethernet/IP packets: 1-to-1 and n-to-1 mapping. 1-to-1 mapping is the simplest strategy that maps each CAN frame into one Ethernet frame. For n-to-1 mapping, proposed GW supports several gateway strategies for mapping multiple CAN frames into one Ethernet frame as proposed in [10]:

- *one-to-one* – reference strategy that maps each CAN frame into one Ethernet frame,
- *buffered* – reduces protocol overhead by waiting for buffer full or timeout events before sending,
- *timed* – extended variant of buffered approach that takes into account CAN frame priority by dynamically reducing the timer value,
- *urgency* – adds a function of instant send to the timed approach to lower the latency.

With the aim to decrease the latency of forwarding high priority CAN frames between domains we propose new strategy:

- *priority* – extends urgency approach with handling priority of messages on the receiving side. The strategy makes use of CPC field from the proposed protocol to determine the priority of received EMESSAGE. When multiple EMESSAGES are present in the receive buffer, the one with lower CPC value will be processed first. In addition, the frames from EDATA are transmitted to the destination bus in order from the highest priority to the lowest.

3.3 Security layer

This Subsection proposes security mechanisms needed to meet security requirements discussed in Section 2. According to our previous analysis of using TCP/IP security protocols in automotive field [13], the best performance results were achieved by IPsec protocol [14]. Therefore it is used to protect backbone communication. IPsec consists of two protocols – Authentication Header (AH) that provides authentication, integrity, replay protection and Encapsulated Security Payload (ESP) that adds also confidentiality. Another advantage is transparency for higher layers and support for central configuration of security parameters as a service in OS or module in middleware which eases implementation of security services and improves manageability and flexibility of the solution. During the design of the security extension we proceeded according to the guidelines described in BCP 146 [15].

3.3.1 Communication security

IPsec protocols and mode. It is crucial to guarantee integrity and authenticity of backbone messages. These services are provided by both AH and ESP. Our previous analysis showed minimal difference in latency between AH and ESP, therefore we suggest that both protocols should be supported to provide confidentiality, if needed. Usage of IPsec in transport mode is proposed as the Gateways are communicating directly with each other and there is no address translation scheme used.

Key management. Manual key management has been chosen because it does not introduce delays unlike automated key management that requires several messages to authenticate endpoints and agree on symmetric key that will be used. Overhead of automatic key management using Internet Key Exchange (IKE) for standard computers is around 370 ms [16].

Identification and authentication. Because of the manual key management the identification of transmitting Gateways is by their IP address. Authentication of Gateways themselves is indirect (only the messages are authenticated using shared secret).

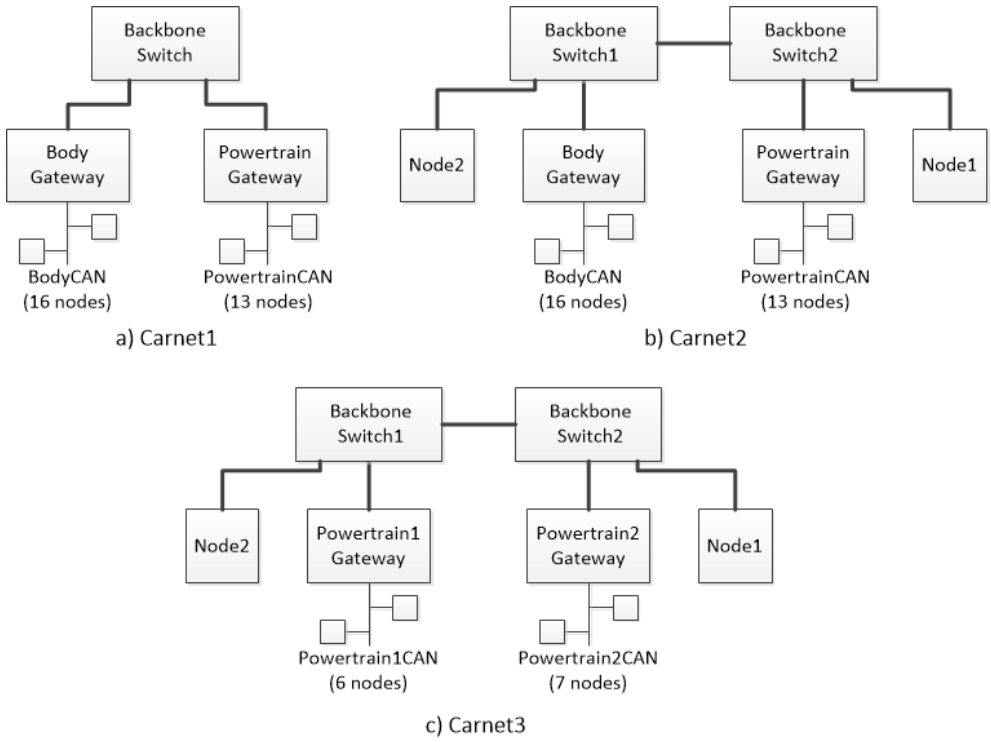


Fig. 4. Experiment topologies

Table 3. Simulation parameters

Element	Parameter	Value
Domain Gateway	CAN proc. delay	206 μs
Domain Gateway	In. processing delay (AH)	108 μs
Domain Gateway	Out. processing delay (AH)	108 μs
Domain Gateway	In. processing delay (ESP)	148 μs
Domain Gateway	Out. processing delay (ESP)	148 μs
GW strategy (excl. 1-to-1)	Buffer length	20
GW strategy (excl. 1-to-1)	Buffer timeout	5 ms
Timed strategy	High prio. threshold	any class0 msg
Urgency/priority strategy	Medium prio. threshold	any class1 msg
Urgency/priority strategy	Instant send threshold	any class0 msg

Table 4. Variables used in the experiment

Variable	Description
Forwarding direction	Carnet1 and Carnet2 topology: direction of forwarded CAN traffic (Body→Powertrain (1) or Powertrain→Body(2))
Gateway strategy	Strategy used for encapsulating CAN frames into UDP segments (one-to-one, buffered, timed, urgency, priority)
Forwarding rate	Percentage of CAN IDs forwarded from CAN to Ethernet. Random IDs from whole network are chosen for each percentage
Background traffic rate	Carnet2 and Carnet3 topology: non-priority background UDP traffic between node1 and node2 (10 - 90 Mbps)
Priority traffic rate	Carnet2 and Carnet3 topology. 1-5 priority traffic streams. Simulates additional communicating Gateways on the backbone. Bandwidth of stream is 1.5 Mbps which corresponds to forwarding all traffic from Powertrain CAN with one-to-one strategy (worst case)
Backbone technology	Carnet2: priority mechanism on the backbone network (Ethernet II, Eth. 802.1p, Eth. AVB). Benchmarked CAN traffic has priority over background traffic: PCP 7 in 802.1p, Class A in Ethernet AVB

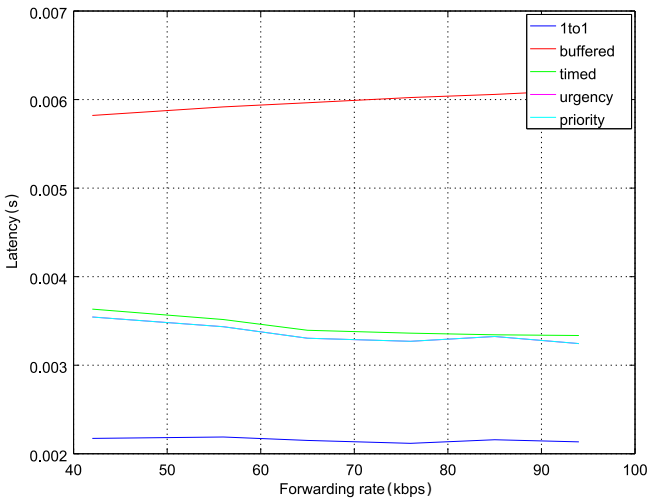


Fig. 5. Effect of CAN forwarding rate (Carnet1 topology, direction $B \rightarrow P$)

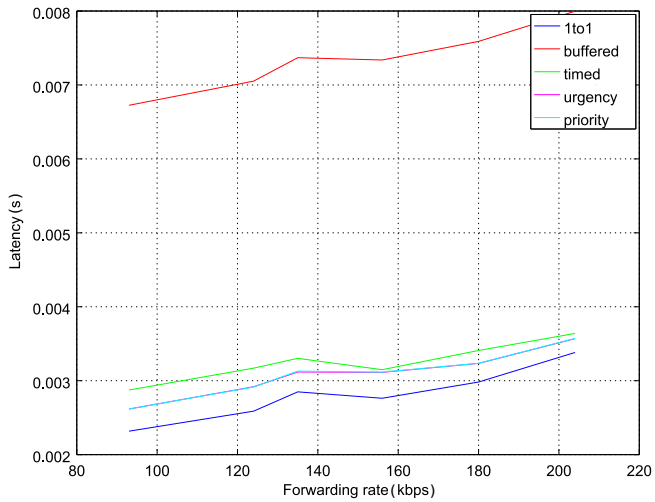


Fig. 6. Effect of CAN forwarding rate (Carnet1 topology, direction $P \rightarrow B$)

Table 5. Utilization based on rate of forwarded traffic

	Forward perc.(%)	50	60	70	80	90	100
Direction	Topology			Carnet1 and Carnet2			
B↑→P	Fw. bitrate (kbps)	42	56	65	76	86	94
B↑→P	Rx utilization (%)	81.5	11.3	13.0	15.4	17.2	18.9
P↑→B	Fw. bitrate (kbps)	93	124	135	156	180	203
P↑→B	Rx utilization (%)	37.2	49.9	54.1	62.4	72.1	81.5
Network	Topology			Carnet3			
1	Fw. bitrate (kbps)	62	71	97	107	116	146
1	Rx utilization (%)	33.6	34.2	35.5	36.0	36.6	40.8
2	Fw. bitrate (kbps)	22	25	31	34	37	58
2	Rx utilization (%)	24.1	25.9	31.1	33.0	34.9	40.8

3.3.2 Access control and secure configuration

In addition to securing the communication, access control mechanism is needed to eliminate the possibility of forwarding inappropriate frames between domains (*eg* sending brake-related frames from Multimedia domain). This can be achieved by using access control lists (ACL) to filter traffic both from the backbone network to local domain network and also vice versa. Aim is to prevent further compromise in case a potential attacker has gained control of an ECU or inserted a malicious node under his control on the local CAN.

Thanks to static nature of in-vehicle network architecture it is possible to apply stricter configuration compared to traditional IP networks. To increase resistance of the backbone network to Man-in-the-Middle attacks we propose that IP addresses and ARP tables of Gateways are statically configured. Furthermore control traffic should be transmitted on a dedicated Virtual LAN (VLAN) with suitable QoS.

4 Evaluation

Designed security extension is evaluated in OMNeT++ simulation environment [17] using INET, CoRE4INET and FiCo4OMNet models [18–20]. In order to simulate real-world CAN example, CAN bus configuration, nodes, and CAN-ID sets are generated using NETCARBENCH software [21].

The objective of the experiment is to determine timing characteristics of the proposed security extension in multiple scenarios and verify if it is possible to use IPsec in transport mode to protect communication on automotive Ethernet/IP backbone. Both AH and ESP are evaluated.

4.1. Network topology

Three network topologies (named Carnet1, Carnet2, Carnet3) have been considered as shown in Fig. 4. The configuration of Controller Area Network domains is the same for Carnet1 and Carnet2. In both cases there are two CAN-Bus networks, BodyCAN (250 kbps bitrate) and PowertrainCAN (500 kbps bitrate). The first two topologies use the same CAN ID set and their purpose is to measure latency of one-way communication between

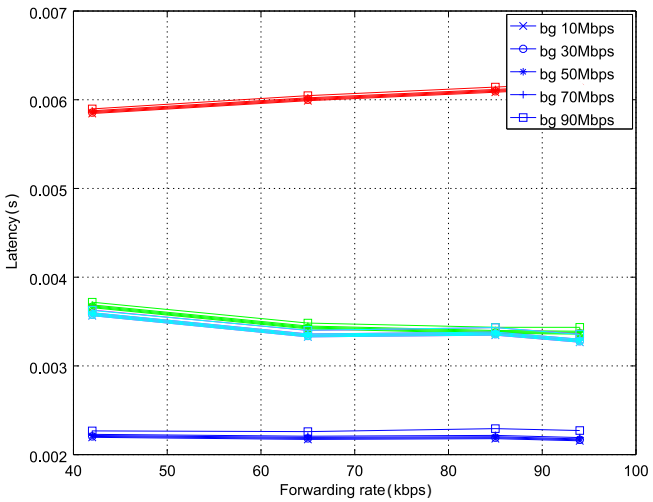


Fig. 7. Effect of low-priority background traffic (Carnet2 topology) The colours represent gateway strategies: blue – one-to-one, red – buffered, green – timed, magenta – urgency, cyan – priority

the domains. The first two topologies differ in backbone configuration. Carnet2 uses two backbone switches and additional nodes (Node1 and Node2) that generate background traffic on the backbone network in order to study the correlation between backbone utilization and latency of benchmarked CAN traffic. Carnet3 topology consists of Powertrain1 and Powertrain2 networks that are made by dividing the ECUs in PowertrainCAN into two sets (to avoid CAN-ID collisions between the forwarded and local frames). It is used to evaluate the latency of two-way communication and differs from the previous two topologies in CAN-Bus configuration.

4.2. Testing method and model parameters

Testing method is based on measuring end-to-end latency of benchmarked CAN traffic – CAN frames forwarded between two domains: Body – Powertrain and Powertrain1 – Powertrain2. CAN frames have been divided into four priority classes (class0 ... class3) based on their CAN identifiers to study the influence of CAN ID on the total latency. The testing consists of several simulation runs that are 30 s long and differ in values of the observed variables. Assignment of simulation parameters is shown in Table 3 and Table 4 shows variables whose effect is studied. Input processing delay and output processing delay are derived from IPsec/AH latency measured in our previous work.

4.3 Experimental results

In this Section, we present results of measurements. Subsections discuss effect of studied variables on the mean latency of benchmarked CAN traffic.

4.3.1 Bitrate and utilization

Measured bitrate of forwarded CAN traffic and utilization of destination CAN network are summarized in Table 5. RX utilization is considerably higher in case of forwarding from Powertrain to Body (direction 2) because of different speeds (500 kbps and 250 kbps).

4.3.2 Effect of CAN traffic forwarding rate

The impact of the amount of CAN traffic forwarded through the Gateway (forwarding rate) on mean latency for different gateway strategies in Carnet1 topology is illustrated in Figures 5 (direction 1: BodyCAN to PowertrainCAN) and 6 (direction 2: PowertrainCAN to BodyCAN). The results for direction 1 meet the expectations. Latency in direction 2 has increasing tendency regardless of the used strategy even if the latency for timed, urgency and priority strategies should decrease thanks to higher frequency of received frames. This is most probably caused by high utilization of BodyCAN due to the fact that its bandwidth is only 250 kbps compared to the 500 kbps on PowertrainCAN. The difference between urgency and priority strategy is negligible as can be seen from overlapped result lines. The impact of CAN traffic forwarding rate in Carnet2 topology is very similar to Carnet1 due to the same traffic pattern. The effect of forwarding rate in Carnet3 topology is exhibiting mean latency almost invariant of forwarding rate for all strategies.

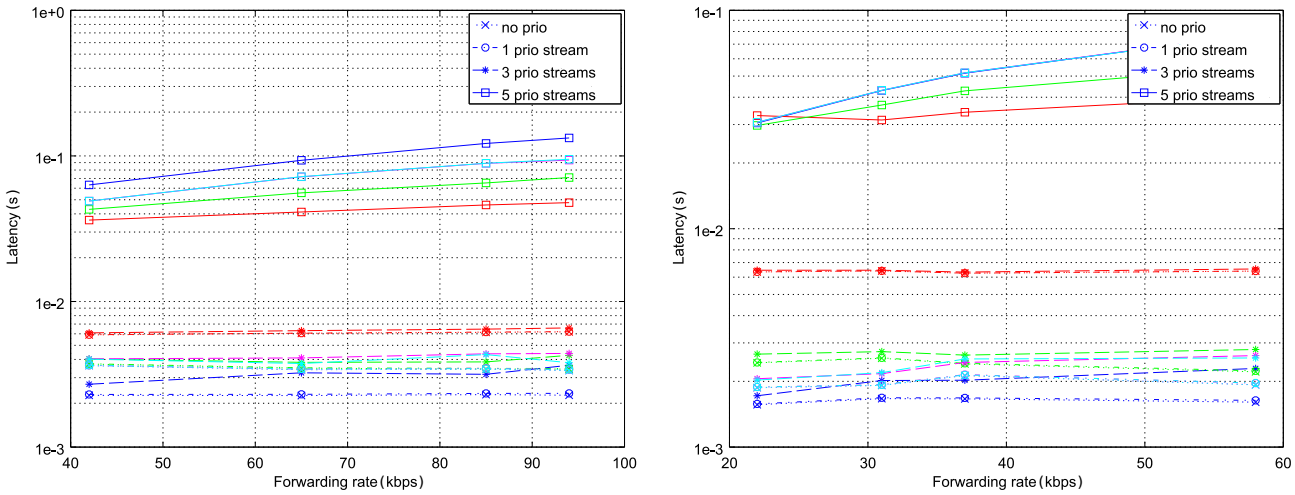
4.3.3 Effect of low-priority background traffic

UDP background traffic flowing through the backbone network has minimal effect on the benchmarked CAN traffic regardless of topology even if no priority is used on the backbone. Figure 7 shows relation between latency and CAN forwarding rate for different background traffic rates. The graph shows mean latency for Carnet2 topology with CAN traffic flowing in direction 1. Line colours represent different forwarding strategies and marker shape indicates the background traffic bitrate. Direction 2 is not included here because it shows the same behaviour as direction 1.

As can be seen from Fig. 7, resulting curves are overlapped which means that latency is practically independent of the low-priority unrelated traffic occurring on the backbone. Therefore there should be no concern about additional traffic on the backbone unless very high bandwidth (over 90 %) is occupied.

4.3.4 Effect of high-priority background traffic

High-priority background traffic on the backbone has almost no influence on the latency in most cases. As the priority streams build up on top of existing non-priority background traffic the situation is different when the backbone is already highly utilized by the non-priority UDP traffic. Figure 8(a) and (b) illustrates the case of 90 Mbps background traffic for Carnet2 and Carnet3.



a) Carnet2 b) Carnet3

Fig. 8. Effect of high-priority background traffic ((a) – Carnet2 vs (b) – Carnet3 topology). The colours represent gateway strategies: blue – one-to-one, red – buffered, green – timed, magenta – urgency, cyan – priority

Line colours represent different forwarding strategies, and markers and lines shapes indicate the number of priority streams flowing in the background. Regardless of forwarding rate, the effect of background traffic is negligible until an attempt is made to transmit 5 priority streams. At this point the backbone has been overloaded (over 98 % utilization) and as a result the latency exceeds timing constraints due to buffering of packets (solid lines in graphs).

This experiment confirms that the priority of background traffic on the backbone has very little influence on the latency, similarly to non-prioritized background traffic. It is, however, important to note that the priority streams occupy much less bandwidth than low-priority traffic. The reason is that it is not expected that high amounts of high-priority traffic will be present on the backbone. It can be assumed that priority traffic would increase the resulting latency faster than low-priority background traffic. The recommendation that follows from the results is not to forget about the bandwidth used by the control traffic streams on the backbone and take it into account during network design to avoid overloading.

4.3.5 Effect of backbone prioritization

Excessive delays during overload situation can be avoided by introducing priority mechanisms implemented on the backbone network level. The effects of two evaluated solutions (Ethernet 802.1p quality of service defined for virtual LANs (VLAN) and Ethernet AVB) have been studied. In both cases the highest possible priority has been assigned to the benchmarked CAN traffic and priority background traffic and lowest priority (no priority tag) to the remaining background traffic. Minimal difference between 802.1p VLAN QoS and Ethernet AVB was observed which means that both technologies are possible candidates for future automotive networks.

4.3.6 Comparison of AH and ESP

Usage of ESP protocol to secure the backbone messages has been simulated on all three topologies (Carnet1, Carnet2, Carnet3) with Ethernet as the backbone technology. The IPsec/ESP latency has almost identical characteristics to IPsec/AH as evaluated in the previous sections. The results of selected simulation runs are shown in the Table 6. As expected from processing delay (Table 4), ESP shows consistent increase in latency by 80 microseconds when compared to AH. The difference is bigger in case of backbone overload (last row in Table 6).

4.3.7 Summary

To sum up the end-to-end latency is influenced by several factors with varying rate. The most significant influence on measured latency is caused by the choice of gateway strategy: one-to-one strategy provides the best performance in terms of latency and jitter but the overhead on the backbone network is the highest. Mean latency of Urgency and Priority strategies is almost identical. The difference is in the latency of high-priority CAN frames where Priority strategy gives approximately 100 μs lower latency, whereas latency of low-prirority frames is higher compared to Urgency strategy. The next significant factor is the rate of CAN traffic that is forwarded between domains. Furthermore, the utilization of destination CAN network and CAN ID dataset have also noticeable impact on the latency. The influence of additional backbone traffic is minimal. Adding confidentiality to the backbone messages by using IPsec/ESP security protocol results in approximately 80 μs higher latency compared to authentication-only communication using IPsec/AH.

Compared to experiments by Kern *et al* [10], our solution exhibits 1.3–1.5 ms higher latency. However. authors in that work do not use any security mechanisms and the destination CAN-Bus has zero utilization (apart

Table 6. Mean latency comparison of AH and ESP (Carnet2 topology, direction $B \rightarrow P$)

		AH mean latency (μs)					ESP mean latency (μs)				
Bg. rate (Mbps)	Prio. rate (streams)	One2one	Buffered	Timed	Urgency	Priority	One2one	Buffered	Timed	Urgency	Priority
Forwarding rate 100 % (94 Kbps)											
10	0	2156	6131	3359	3267	3267	2237	6212	3440	3348	3348
10	5	2158	6133	3360	3269	3269	2239	6214	3442	3350	3350
30	0	2167	6142	3369	3278	3278	2248	6223	3450	3359	3359
30	5	2172	6146	3373	3282	3282	2253	6228	3454	3364	3363
50	0	2177	6151	3379	3288	3289	2259	6233	3460	3370	3370
50	5	2185	6158	3386	3295	3295	2267	6239	3467	3376	3376
70	0	2192	6163	3391	3301	3301	2275	6244	3473	3382	3382
70	5	2207	6171	3403	3313	3313	2290	6253	3483	3394	3395
90	0	2272	6187	3435	3359	3355	2355	6268	3522	3434	3440
90	5	132881	47762	71034	93643	94852	139102	48924	73016	99564	99124

from incoming traffic from the GW). Our results exhibit security processing overhead comparable to software security mechanisms implemented directly on Controller Area Network [5], however the total latency in our case is considerably higher because inter-domain traffic is involved. Hardware-based measurements [7] used different hardware and only traffic on one CAN-Bus was analysed. Nevertheless the latency is similar to our findings.

5 Conclusion

In this work we present a security extension of automotive communication protocols that uses Ethernet/IP technology. The solution is based on encapsulation of automotive frames into UDP datagrams with added authenticity, integrity and (if required) confidentiality of communication using IPsec protocol in transport mode. Proposed method has been implemented and evaluated for Controller Area Network which is currently the most widespread automotive bus technology. It has been tested in simulation environment with configuration based on experiments on real hardware. The results indicate that our solution based on IPsec provides suitable performance and security properties for use in domain-based automotive embedded networks. The stated 10ms hard deadline was violated during simulation when Ethernet backbone had been overloaded but in "normal" backbone conditions, the average end-to-end latency spans approximately from 1.5 ms to 8 ms.

Advantages of the solution are no limitation to the length and content of CAN frames as opposed to security protocols directly protecting CAN frames, compatibility with current technologies that enables implementation of security measures without need to alter the CAN ID sets of existing systems, possibility to incorporate IPsec service into embedded OS or middleware of ECUs. Moreover, thanks to using IP-based communication, the concept is compatible with Car2X communication which is also one of topics planned in our future research.

The contribution of this work is the development of a novel approach to secure the control communication in automotive systems that takes advantage of emerging Ethernet/IP applications in vehicles and proven security solutions from TCP/IP communication model, notably IPsec protocol. Presented approach proposes extensible method to encapsulate automotive bus frames into IPsec-protected datagrams. Another contribution is experimental evaluation and detailed benchmarking of IPsec performance for different scenarios of forwarding CAN frames between in-vehicle CAN domains interconnected by Ethernet backbone. Results of the presented method indicate that implementation of IPsec protocol support in automotive embedded operating systems would be beneficial to improve the security of communication in modern and future "connected" vehicles.

Acknowledgement

This work was partially supported by Eset Research Centre, research grant VEGA 1/0836/16 "Methods and algorithms for effective and reliable delivery of multimedia content, and Research and Development Operational Programme for the project "University Science Park of STU Bratislava, ITMS 26240220084, co-funded by the European Regional Development Fund.

References

[1] I. Studnia, V. Nicomette, E. Alata, Y. Deswarte, M. Kaniche and Y. Laarouchi, "Survey on Security Threats and Protection Mechanisms in Embedded Automotive Networks," 2013 43rd Annual IEEE/IFIP Conference on Dependable Systems and Networks Workshop (DSN-W), Budapest, Hungary, 2013, pp. 1–12, doi: 10.1109/DSNW.2013.6615528.

[2] S. Checkoway., D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner and T. Kohno, "Comprehensive Experimental Analyses of Automotive Attack Surfaces", in Proc. 20th USENIX Conference on Security, USENIX Association, Berkeley, CA, USA, 2011.

[3] S. Woo, H. J. Jo and D. H. Lee, "A Practical Wireless Attack on the Connected Car and Security Protocol for In-Vehicle CAN,"

in IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 2, pp. 993–1006, April 2015, doi: 10.1109/TITS.2014.2351612.

[4] Robert Bosch GmbH, "CAN Specification Version 2.0", 1991.

[5] J. A. Bruton, "Securing CAN Bus Communication: An Analysis of Cryptographic Approaches", National University of Ireland, Galway, Ireland, 2014.

[6] M. Chavez, C. Rossete and F. Henriquez, "Achieving Confidentiality Security Service for CAN", in Proc. 15th Int. Conf. Electronics, Communications and Computers, 2005 pp. 166–170.

[7] H. Schweppe, "Security and Privacy in Automotive On-Board Networks", Networking and Internet Architecture [cs.NI] Télécom ParisTech, Paris, France, 2012.

[8] K. Beretis and I. Symeonidis, "Experimental Evaluation of End-to-End Delay in Switched Ethernet Application in the Automotive Domain", in SAFECOMP 2013 – Workshop CARS (2nd Workshop on Critical Automotive applications: Robustness & Safety) of the 32nd International Conference on Computer Safety, Reliability and Security, Toulouse, France, 2013.

[9] H.-T. Lim, L. Volker and D. Herrscher, "Challenges in a Future IP/Ethernet-Based In-Car Network for Real-Time Applications", in Proc. 48th ACM/EDAC/IEEE Design Automation Conference (DAC), 2011, pp. 7–12.

[10] A. Kern, D. Reinhard, T. Streichert and J. Teich, "Gateway Strategies for Embedding of Automotive CAN-frames into Ethernet-packets and Vice Versa", in Proc. 24th Int. Conf. on Architecture of Computing Systems, 2011, pp. 259–270.

[11] P. Hank, S. Müller, O. Vermesan and J. Van Den Keybus, "Automotive Ethernet: In-Vehicle Networking and Smart Mobility", in 2013 Design, Automation & Test in Europe Conference & Exhibition (DATE), 2013, pp. 1735–1739.

[12] "SocketCAN", 2021, https://www.kernel.org/doc/Documentation/networking.can.txt [Accessed: 26-Feb-2021].

[13] J. Lastinec and L. Hudec, "Comparative Analysis of TCP/IP Security Protocols for Use in Vehicle Communication", in Proc. 17th International carpathian control conference, Tatranská Lomnica, Slovak Republic, 2016.

[14] S. Kent and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301 (Proposed Standard), IETF, 2005.

[15] S. Bellovin, "Guidelines for Specifying the Use of IPsec Version 2", BCP146, IETF, 2009.

[16] C. Shue, Y. Shin, M. Gupta and J. Y. Choi, "Analysis of IPSec Overheads for VPN Servers", in 1st IEEE ICNP Workshop on Secure Network Protocols (NPSec), 2005, pp. 25–30.

[17] "OMNet++", 2021. [Online]. http://omnetpp.org. [Accessed: 26-Feb-2021].

[18] "INET Framework", 2021. [Online]. http://inet.omnetpp.org. [Accessed: 26-Feb-2021].

[19] S. Buschmann, T. Steinbach, F. Korf and T. C. Schmidt, "Simulation-based Timing Analysis of FlexRay Communication at System Level", in Proc. 6th International ICST Conf. on Simulation Tools and Techniques (SimuTools '13), Brussels, Belgium, 2013, pp. 285-290.

[20] T. Steinbach, H. D. Kenfack, F. Korf and T. C. Schmidt, "An Extension of the OMNeT++ INET Framework for Simulating Real-time Ethernet with High Accuracy", in SIMUTools 2011 – 4th International OMNeT++ Workshop, 2011, pp. 375–382.

[21] C. Braun, L. Havet and N. Navet, "NETCARBENCH: a Benchmark for Techniques and Tools Used in the Design of Automotive Communication Systems", in Proc. 7th IFAC Int. Conf. Fieldbuses & Networks in Industrial & Embedded Systems (FeT 2007), Toulouse, France, 2007.

Received 15 February 2021

Jan Lastinec received master's and doctoral degrees from the Faculty of Informatics and Information Technologies of Slovak University of Technology in Bratislava where he is currently working as a postdoctoral research assistant and teacher. His research interests include security of automotive networks, security monitoring, and cyber range platforms for research and education. He is involved in teaching of network security, security of information technologies, and operating systems.

Ladislav Hudec (doc, Ing, CSc), currently associate professor of computer science and engineering, deputy director of the Institute of Computer Engineering and Applied Informatics, Faculty of Informatics and Information Technologies, Slovak University of Technology in Bratislava. He is author or co-author over 60 scientific papers published in journals and proceedings of the conferences and over 80 technical papers in the field of fault tolerant computing, embedded systems, and information security. He led over 90 research grants and industrial projects. He reads lectures on Principles of information security, Security of information technologies and Internet security. Dr. Hudec is member of the Information Systems Audit and Control Association (ISACA), holder of the CISA license. During the period 1993-2010 he served as national coordinator at the European Cooperation in Science and Technology (COST).