



DoS Protection through Credit Based Metering - Simulation-Based Evaluation for Time-Sensitive Networking in Cars

Philipp Meyer, Timo Häckel, Franz Korf, and Thomas C. Schmidt

Department of Computer Science

Hamburg University of Applied Sciences, Germany

{philipp.meyer, timo.haekkel, franz.korf, t.schmidt}@haw-hamburg.de

Abstract

Ethernet is the most promising solution to reduce complexity and enhance the bandwidth in the next generation in-car networks. Dedicated Ethernet protocols enable the real-time aspects in such networks. One promising candidate is the IEEE 802.1Q Time-Sensitive Networking protocol suite. Common Ethernet technologies, however, increase the vulnerability of the car infrastructure as they widen the attack surface for many components. In this paper we propose an IEEE 802.1Qci based algorithm that on the one hand, protects against DoS attacks by metering incoming Ethernet frames. On the other hand, it adapts to the behavior of the Credit Based Shaping algorithm, which was standardized for Audio/Video Bridging, the predecessor of Time-Sensitive Networking. A simulation of this proposed Credit Based Metering algorithm evaluates the concept.

1 Introduction

In today's vehicles, a multitude of sensors, actuators, and electronic control units (ECUs) are used to enable enhanced performance, comfort, and safety through advanced driver assistance systems. Even autonomous driving will be realized in future generations. These additions result in complex communication over different proprietary bus technologies in multiple domains.

Ethernet technologies are used to set up efficient and straightforward communication in future generations. Real-time Ethernet protocols enable the compliance of communication requirements and enhance the reliability of Standard Ethernet. Promising candidates are the Time-Sensitive Networking (TSN) protocols by the IEEE (<https://1.ieee802.org/tsn/>).

The main focus of those protocols is Quality of Service guarantees. The integration of future cars in the IoT context opens its systems to global communication. These online capabilities and the domain interconnection increase the attack surface of safety-critical functions like brakes and the motor control units. Attacks could manipulate driving characteristics and could provide fatal consequences for vehicle and passengers.

Therefore, security has to be an essential goal for the development of the next generations of on-board communications technologies. The TSN standard IEEE 802.1Qci addresses some security concerns by filtering ingress traffic on network node ports.

This work provides a simulation-based evaluation of IEEE 802.1Qci with a Credit Based Meter (CBM) algorithm concept. This concept enforces the reserved bandwidth of a stream and is one solution to protect network nodes from Denial of Service (DoS) attacks. IEEE 802.1Qci and the CBM concept are implemented in the OMNeT++ environment to enable evaluations.

This paper is organized as follows: Section 2 presents previous and related work. In Section 3, the developed simulation environment, and the credit based metering are shown. Section 4 presents a case study followed by an evaluation of the implemented simulation models and concepts. The paper closes with a conclusion and outlook in section 5.

2 Background & Related Work

The on-board network of a vehicle is a highly distributed system defined by its electronic control units (ECUs). At present, proprietary bus technologies (CAN, MOST, FlexRay) enable the communication of control units.

Future development will lead to a stepwise transition towards flat, switched Ethernet networks [6]. Such networks have to support simultaneous transmission of messages with different priorities to maintain safe time-critical communication. Real-time Ethernet protocols are used to guarantee the different quality of service classes.

The Time-Sensitive Networking (TSN) [5] real-time Ethernet standard is a collection of protocols which adapts to network requirements of cyber-physical systems. Domains for this protocol suite are, for example, industrial control facilities or in-car networks.

The focus in this work is the sub-standard IEEE 802.1Qci [4]. It describes an ingress control through per-stream filtering and policing. Figure 1 shows the structure of the filtering and policing specification. The shown mechanism has an instance after each port ingress of a TSN networking device. The result is that all ingress traffic is filtered. There are three stages an incoming frame has to pass through before it is queued. The first stage consists of a set of stream filters. They configure which gates and meters are responsible for handling frames of a specific stream id. Secondly, there are stream gates. Those have one of the two states "OPEN" or "CLOSED". This state can change based on a static defined schedule based on a system-wide clock. If the gate is "CLOSED", the frame will be dropped. If the responsible gate is in the state "OPEN", the frame will be handled by the responsible flow meter. The flow meters stage contain unique algorithms to assert if a message is allowed. After the meter allows a frame, it gets queued in the network node for subsequent forwarding or processing. This work presents a meter concept called Credit Based Meter (CBM) (See section 3).

The importance of security measures for in-car networks is shown in various related work [2], [10]. A fundamental work is from Checkoway et al. [1]. They examine interfaces that are part of the attack surface in a car. These interfaces are classified in three categories: Physical access (ODB-II, CD, USB), short distance wireless access (Bluetooth, WiFi, Remote-Keyless-Entry) and long distance wireless access (GPS, digital radio, mobile services). The authors gained access to the on-board network in each category using reverse engineering and debugging.

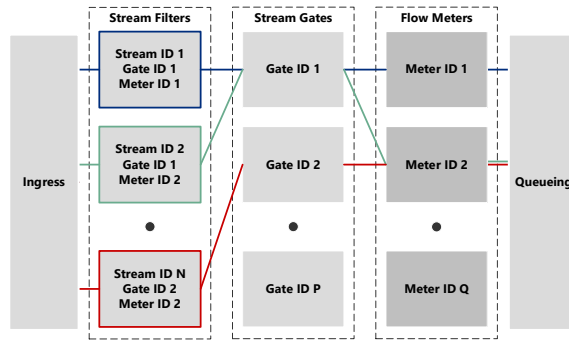


Figure 1: IEEE 802.1Qci per-stream filtering and policing

Miller and Valasek [9] described in detail how to obtain control over an unaltered passenger vehicle. They gained control over safety-critical elements like engines and brakes over a remote cellular connection of the infotainment system. The infotainment system is part of the internal CAN-based communication infrastructure, and CAN buses in this infrastructure contain virtually no security measures. At this point, the authors used reverse engineering to get information over the communication between in-car control units. In the next step CAN messages are forwarded over the infotainment system into the internal communication which is received and processed by the control units.

In consequence, security measures must be included in future automotive communication systems. Simulations of in-car networks [11] are an essential method to study the behavior of such systems in detail. This work presents one solution for protecting the in-car communication from DoS attacks and analyses it in the simulator. In those scenarios, a compromised stream is used to burst frames into networking devices to overload their capabilities. The result could be lost or delayed frames of not compromised streams.

3 Credit Based Metering

The Credit Based Meter (CBM) is an ingress counterpart for the Credit Based Shaper (CBS) egress behavior defined in IEEE 802.1Qbv [3]. In general, a CBM is based on a credit value manipulated by two different gradients called "idleslope" and "sendslope". Frame reception is allowed when the credit is greater or equal to zero. When the credit is lower than zero, an incoming frame will be discarded.

$$idleslope = RB \quad (1)$$

$$sendslope = RB - B \quad (2)$$

These gradients are composed of an accumulated reserved bandwidth (RB) of the streams passing the meter and a total bandwidth (B) of the port (See equation 1 and 2).

Additionally, the CBM contains a maximum burst size parameter ($Burst_{max}$) configuring the maximum count of frames that are allowed in an incoming stream burst. This is used in combination with the sending duration ($T_{duration}$) of one frame. $T_{duration}$ is composed of the frame size (FS_{stream}), the port bandwidth (B) and the Ethernet inter frame gap (T_{ifg}) to calculate the maximum credit value ($Credit_{max}$) of the CBM shown in equations 3 and 4.

$$T_{duration} = \frac{FS_{stream}}{B} + T_{ifg} \quad (3)$$

$$Credit_{max} = |sendslope| * T_{duration} * (Burst_{max} - 1) \quad (4)$$

Because a burst of one frame is allowed when the credit is 0 $Burst_{max}$ has to be subtracted by one. So the definition of $Burst_{max} = 1$ results in $Credit_{max} = 0$.

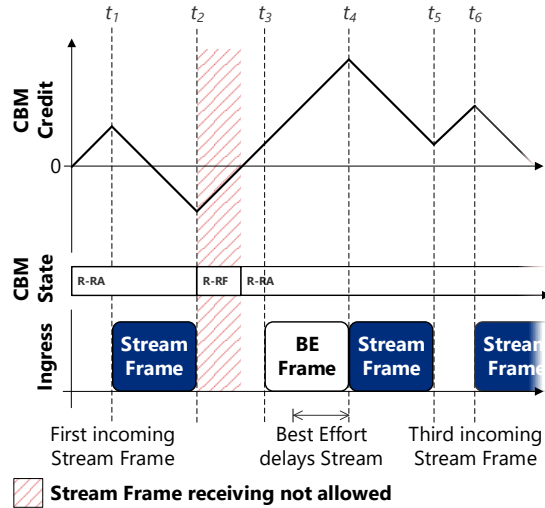


Figure 2: Credit Based Metering state machine

The CBM has two states. They are "RUNNING RECEIVING ALLOWED" (R-RA) and "RUNNING RECEIVING FORBIDDEN" (R-RF). When the CBM starts the state is R-RA and the credit is set to zero. The credit starts to increase according to *idleslope* till the first frame is incoming or the credit reaches the maximum ($Credit_{max}$).

In the R-RA state, the credit is decreased by "*sendslope*" for the receiving duration of a frame. When the frame is queued the credit increases again with "*idleslope*". If the credit is greater or equal to zero, a new frame reception is allowed, and the credit decreases again by "*sendslope*". When the credit reaches the maximum, it stays on this value until a frame is incoming.

If the credit is lower than 0, the state will be switched to R-RF. In R-RF each incoming frame will be deleted. Simultaneously, the credit increases with "*idleslope*". The state is changed back to R-RA when the credit reaches 0.

In figure 2, an example of the CBM algorithm behavior is shown. Firstly, the state is R-RA and the credit is 0 and increases according to "*idleslope*" until the first frame arrives (see t_1 in figure 2). The credit decreases by "*sendslope*" for the duration of the frame (t_2 in figure 2). Now the state is changed to R-RF and the credit increases by "*idleslope*" till it reaches 0. The state changes to R-RA, and the credit increases further until the next frame arrives. This is delayed by an incoming best effort (BE) frame (t_3 in figure 2). The next frame arrives and the credit is decreased again (t_4 to t_5 in figure 2). The credit increases till the third frame receiving starts (t_6 in figure 2). So again the credit decreases by "*sendslope*" until the end of the transmission duration.

The performance of the CBM is dependent on $Burst_{max}$. A target configuration of this parameter is as low as possible and still supports a valid worst-case scenario. On the one side, this is because of the counterpart CBS. The valid maximum frame burst of a stream that is produced by a CBS algorithm egress is dependent on its specific worst-case scenario. On the other side, an attack creating a maximum frame burst could not harm the network because it is designed to support the worst-case traffic workload.

There are different ways to determine a minimal $Burst_{max}$ value. One example is analyzing the worst-case burst behavior for each streams output port ($Burst_{out}$). $Burst_{max}$ has to be calculated, as shown in equation 5 to allow one closeup frame following the burst.

$$Burst_{max} = Burst_{out} + 1 \quad (5)$$

Another example of determining a $Burst_{max}$ value is by simulating different configurations to find one that fits the requirements.

4 Case Study

This section evaluates the integration of the Credit Based Meter algorithm inside IEEE 802.1Qci. This is done by using the OMNeT++ (<https://omnetpp.org/>) simulation environment with INET (<https://inet.omnetpp.org/>) and our CoRE4INET framework. CoRE4INET enables in-car network simulations [7] and the simulations of TSN features [8]. For this work, the CoRE4INET is extended with IEEE 802.1Qci and CBM implementations.

The chosen topology is known from previous work [8] and is designed to create critical links with multiple concurrent traffic. Figure 3 shows this topology. In this topology, time-triggered traffic is based on TDMA with the highest priority. Two configurations are simulated. The first is a configuration with active CBM filtering. The second one replaces a compromised "Node 1" into the simulation, which is spamming a DoS attack into the network.

For both simulations the base configuration is as follows:

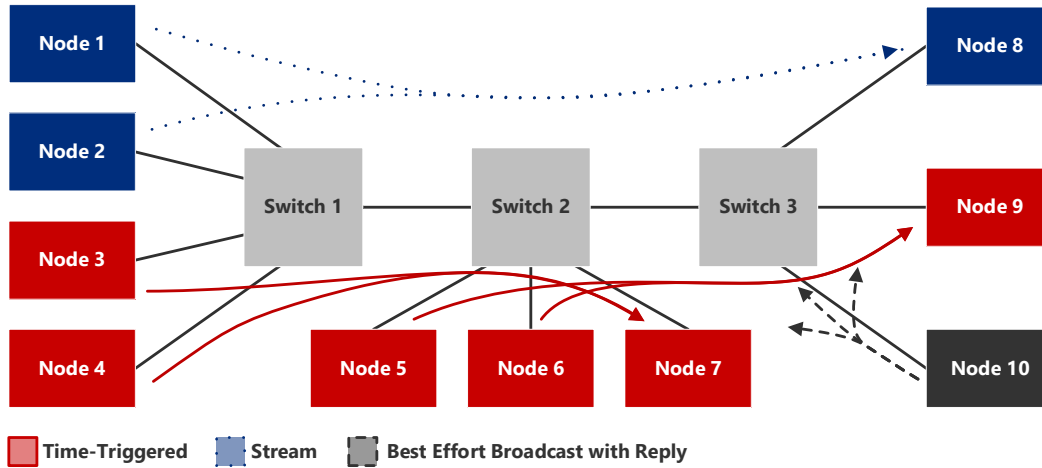


Figure 3: Simulation Topology

- All links are configured with a bandwidth of 100 Mbit/s.
- "Node 1" and "Node 2" are the sources of "Stream 1" and "Stream 2" with "Node 8" as its destination. Both streams have a reserved route with an individual bandwidth of 25 Mbit/s. Each stream is passing a CBM on all devices and gates are "OPEN".
- Full-size time-triggered frames are generated by "Node 3", "Node 4", "Node 5" and "Node 6". The first two are received by "Node 7" and the latter by "Node 9". In all switches, a gap of 123 μ s is configured to allow intermediate frame bursts.
- For extra background traffic "Node 10" is broadcasting full-size best-effort Ethernet frames. All nodes are replying by sending a full-size best-effort frame back to "Node 10".

The worst-case output stream burst sizes ($Burst_{out}$) are known in this base configuration. They are 2 for "Node 1" and "Node 2" and, because of the concurrent TDMA traffic 4 for "Switch 1" and "Switch 2". Therefore the $Burst_{max}$ value for CBM filtering in "Switch 1" is 3 for both input ports and 5 for the input metering in "Switch 2" and "Switch 3" (See equation 5 in section 3).

The results shown in this section are a selection of results generated by the simulations. All shown simulation results are based on 10 seconds duration runs.

The first result set (Figure 4 and 5) presents and compares the end-to-end latency of the streams in both configurations. Due to the assumption that valid packets are not influenced by the CBM, these latencies are expected to be nearly the same.

The end-to-end latency of both streams in the configuration with CBM filtering is shown in figure 4. In this configuration, each node and switch is using a CBM ingress control on each port and for each stream. The two histograms show the number of frames that arrived at the target with a specific consolidated end-to-end latency. Blue shows these results for "Stream 1" and grey for "Stream 2".

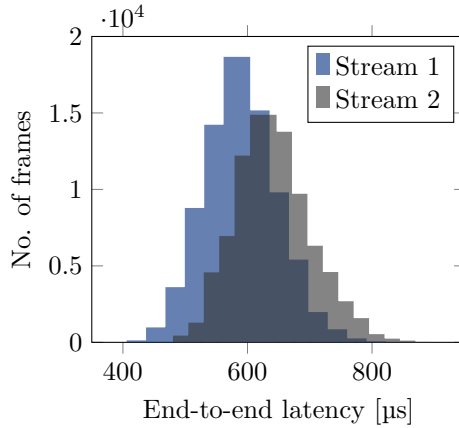


Figure 4: End-to-end latency of frames per stream without attack

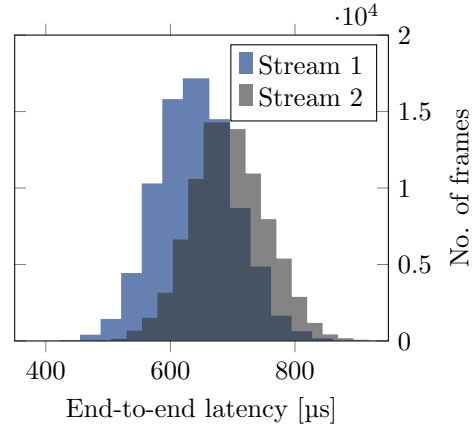


Figure 5: End-to-end latency of frames per stream during an attack

Figure 5 shows the end-to-end latency of the streams in a configuration where "Node 1" is corrupted. All nodes and switches are using CBM ingress control again. The difference is that "Node 1" is generating the "Stream 1" packets in an invalid pattern. This is done by spamming subsequent frames.

The comparison shows no significant differences between the configurations. This demonstrates that CBM is successfully enforcing the correct behavior. This is done by removing all "Stream 1" frames of the corrupted source that would exceed the reserved bandwidth. Therefore "Stream 2" is not affected by "Node 1" spamming.

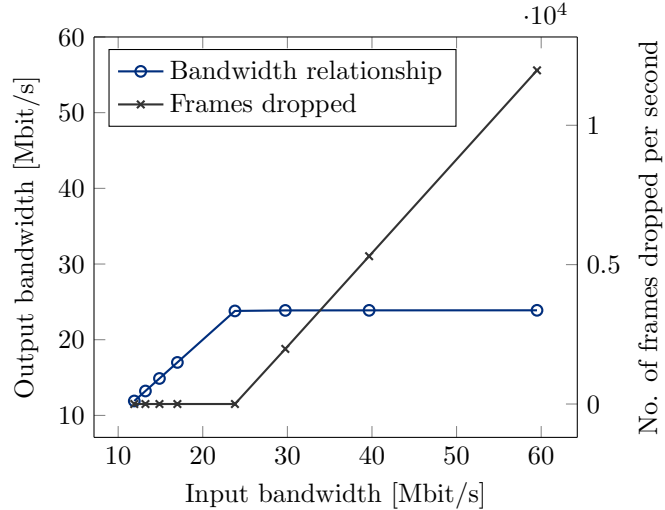


Figure 6: Impact of CBM on Stream 1 in Switch 1

Figure 6 presents output bandwidth size and number of frames dropped in the CBM in "Switch 1" for "Stream 1" produced by eight simulation runs. For each run, the input bandwidth produced by "Node 1" is incremented. The reserved bandwidth of 25 Mbit/s is fixed. It is expected that the output would not cross this reserved bandwidth value.

The result reflects the wanted CBM behavior. No frame is dropped, and the output bandwidth is the same as the input bandwidth until the input size overshoots the reserved bandwidth of 25 Mbit/s. At this point, the number of frames dropped increases as a function of the input bandwidth. Because each frame, which would exceed the reserved bandwidth, will be dropped by the CBM.

A selected section of this CBM algorithm is shown in Figure 7. It presents the credit value, frame ingress, and output bandwidth for a specific timeslot of the simulation. "Node 1" produces a valid "Stream 1" packet flow of 25 Mbit/s in this scenario. Although the CBM

output bandwidth never exceeds the reserved bandwidth over time, it allows short crossings like its counterpart CBS.

Because no frame is received between 146.625 ms and 146.75 ms the credit increases as a function of this duration. This continues until it reaches its maximum, which is dependent on $Burst_{max}$.

In this case, the $Burst_{max}$ value is 3. This results in a $Credit_{max}$ value of ca. 4650. The corresponding equation 6 shows the calculation of this $Credit_{max}$ value.

Next, a continuous burst of 3 frames would be allowed. In this case, just two subsequent packages are incoming. This results in a zoomed in bandwidth of 50 Mbit/s between 146.75 ms and 146.875 ms.

This shows that the reserved bandwidth could be overshoot massively for shorter periods. This is dependent on the configuration of $Burst_{max}$. From this also follows that $Burst_{max}$ value has no influence on the over-time bandwidth restriction. Buffer sizes have to support the $Burst_{max}$ values to guarantee that they did not overflow.

The CBM enforces this upper barrier. For a configured network, the maximum latencies could be calculated and are valid even if a malfunction or attack results in an invalid behavior of individual network participants. This protects the integrity and availability of the in-car communication system.

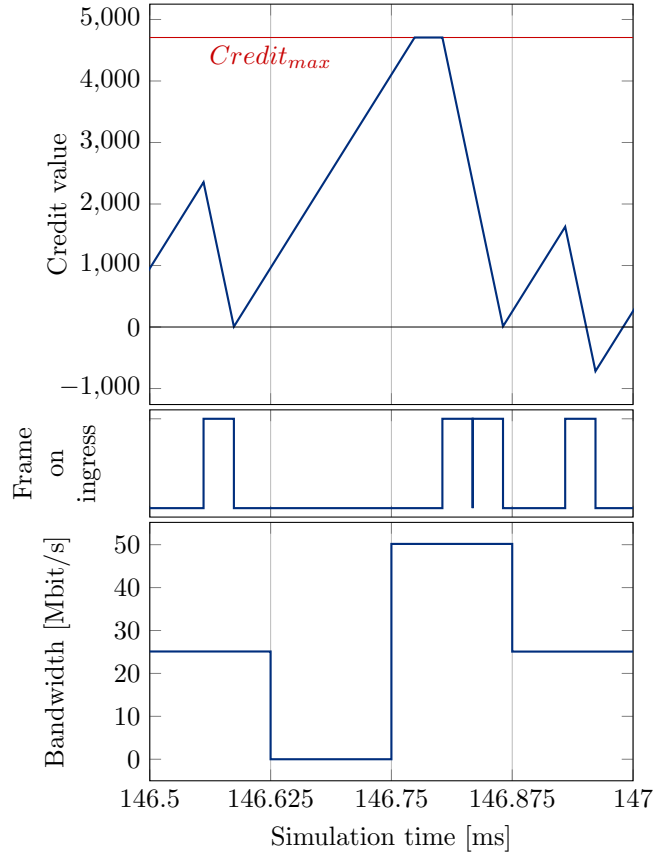


Figure 7: Section of CBM credit, frame, and bandwidth

$$\begin{aligned} Credit_{max} &= |sendslope| * T_{duration} * (Burst_{max} - 1) \\ &\approx 75 \text{ Mbit/s} * 31 \mu\text{s} * (3 - 1) = 4650 \end{aligned} \quad (6)$$

5 Conclusion & Outlook

The demand for interconnecting an increasing multitude of sensors, actors, and ECUs in today's vehicles guides in-car networks to adapt real-time Ethernet technologies. Flattening the network in this way creates new vulnerabilities within the in-car network. The CBM is a solution for a TSN meter algorithm to protect the system against DoS attacks. It protects the integrity and availability of an in-car communication system by individually controlling the stream input on each ingress port of the network. The CBM allows all valid traffic patterns of a CBS algorithm. An attacker could use the burst behavior to shortly overcome the reserved bandwidth

restrictions, but the credit boundary limits the bandwidth over an extended period. This limit is the same as the reserved bandwidth. The maximum burst parameter has to be as low as possible to gain the best performance. However, it still must allow the valid worst-case scenario of a specific input port. This trade-off between performance and worst-case estimation has to be considered.

In future work, the compatibility with other TSN traffic shaper concepts will be evaluated. Furthermore combined operation of different ingress control mechanisms will be simulated. In addition, the benefits of the ingress control metrics for anomaly detection will be analyzed.

Our simulation models and the extensions in this work are published at sim.core-rg.de.

Acknowledgments

This work is funded by the German Federal Ministry of Education and Research (BMBF) within the SecVI project.

References

- [1] Stephen Checkoway, Damon Mccoy, Brian Kantor, Danny Anderson, Hovav Shacham, Stefan Savage, Karl Koscher, Alexei Czeskis, Franziska Roesner, and Tadayoshi Kohno. Comprehensive Experimental Analyses of Automotive Attack Surfaces. *USENIX Security*, 2011.
- [2] K. Iehira, H. Inoue, and K. Ishida. Spoofing attack using bus-off attacks against a specific ecu of the can bus. In *2018 15th IEEE Annual Consumer Communications Networking Conference (CCNC)*, pages 1–4, Jan 2018.
- [3] Institute of Electrical and Electronics Engineers. IEEE Standard for Local and metropolitan area networks – Bridges and Bridged Networks - Amendment 25: Enhancements for Scheduled Traffic. Standard, IEEE, March 2016.
- [4] Institute of Electrical and Electronics Engineers. IEEE Standard for Local and metropolitan area networks–Bridges and Bridged Networks–Amendment 28: Per-Stream Filtering and Policing. Standard, IEEE, September 2017.
- [5] Institute of Electrical and Electronics Engineers. IEEE Standard for Local and Metropolitan Area Network–Bridges and Bridged Networks. Standard, IEEE, July 2018.
- [6] Kirsten Matheus and Thomas Königseder. *Automotive Ethernet*. Cambridge University Press, Cambridge, United Kingdom, January 2015.
- [7] Philipp Meyer, Franz Korf, Till Steinbach, and Thomas C Schmidt. Simulation of mixed critical in-vehicular networks. In *Recent Advances in Network Simulation*, pages 317–345. Springer, 2019.
- [8] Philipp Meyer, Till Steinbach, Franz Korf, and Thomas C. Schmidt. Extending IEEE 802.1 AVB with Time-triggered Scheduling: A Simulation Study of the Coexistence of Synchronous and Asynchronous Traffic. In *2013 IEEE Vehicular Networking Conference (VNC)*, pages 47–54, Piscataway, NJ, USA, December 2013. IEEE Press.
- [9] Charlie Miller and Chris Valasek. Remote exploitation of an unaltered passenger vehicle. *Black Hat USA*, 2015:91, 2015.
- [10] Syed Rizvi, Jonathan Willett, Donte Perino, Tyler Vasbinder, and Seth Marasco. Protecting an automobile network using distributed firewall system. In *Proceedings of the Second International Conference on Internet of Things, Data and Cloud Computing, ICC '17*, pages 174:1–174:6, New York, NY, USA, 2017. ACM.
- [11] Till Steinbach, Hermand Dieumo Kenfack, Franz Korf, and Thomas C. Schmidt. An Extension of the OMNeT++ INET Framework for Simulating Real-time Ethernet with High Accuracy. In *Proceedings of the 4th International ICST Conference on Simulation Tools and Techniques*, pages 375–382, New York, March 2011. ACM-DL.