

Preventing DoS Attacks in Time Sensitive Networking In-Car Networks through Credit Based ingress Metering

Philipp Meyer

philipp.meyer@haw-hamburg.de

Department of Computer Science, Hamburg University of Applied Sciences, Germany

Abstract—Ethernet is the most promising solution to reduce complexity and enhance the bandwidth in the next generation in-car networks. The real-time aspects in such networks are becoming possible through special Ethernet protocols. On promising candidate is the IEEE 802.1 Time Sensitive Networking protocol suite. However, the common Ethernet technology increases the vulnerability of the cars infrastructure. In this paper an algorithm is proposed which on the one hand provide protection against DoS attacks on switches by metering incoming Ethernet frames. And on the other hand adapts to the behaviour of the Credit Based Shaping algorithm which was firstly standardized in the Time Sensitive Networking predecessor Audio/Video Bridging. The evaluation of this proposed Credit Based Metering algorithm is evaluated on a state machine implementation.

I. INTRODUCTION

In todays vehicles a multitude of sensors and electronic control units (ECUs) are used to enable better performance, comfort and safety. They are used to enable advanced driver assistance systems and autonomous driving. This results in complex communication architectures containing different proprietary bus technologies.

With the use of Ethernet as a backbone in future cars the architecture could be simple and efficient. Real-time Ethernet protocols enable the compliance of communication requirements and enhance the reliability of Standard Ethernet. One promising candidate is the Time Sensitive Networking (TSN) protocol which is in the process of standardization through IEEE.

The current focus in those protocols is safety. The integration of future cars in the IoT context opens its systems to global communication. This increases the vulnerability of the infrastructure and safety critical functions like brakes or the motor control unit resulting in manipulations of the driving characteristics. The Results could provide fatal consequences for vehicle and passengers.

Therefore security has to be a major goal for the development of the next generation on-board communication technologies.

This work provides a concept for the protection of queues in a TSN switch from DoS attacks. Those queues output is handled by the so called Credit Based Shaper (CBS). But there is no system that controls the input is matching the configured output. Therefore the queues in switches could be filled to prevent other messages from reaching the output destination.

Here the concept introduces a Credit Based Metering (CBM) algorithm to control the queues input flow. Its a syntactic approach to prevent that attack independent streams are affected. So a safety relevant stream can not be blocked by not safety relevant streams.

This paper is organised as follows: Section II presents previous and related work. In Section III the concepts of security, in-car networks and the credit based shaping are made known. Section IV introduces the attack scenario and the Credit Based Metering protection concept. Section V provides the evaluation of the CBM concept. Finally, Section VI concludes the work and gives an outlook on future research.

II. RELATED & PREVIOUS WORK

In previous work the focus is analysing the safety aspects of combining synchronous and asynchronous traffic in a in-car Time Sensitive Networking architecture [1]. A simulation based analysis of the impact of TDMA traffic on Audio Video Bridging streams is discussed. In contrast this work will analyse security aspects of the asynchronous part of Time Sensitive Networking in-car networks.

Checkoway et al. [2] examine wich interfaces are attackable in an automobile. This interfaces are classified in three categories. The first one is indirect physical access. It includes ODB-II diagnose interface and the infotainment ports (CD, USB). The next category is called short distance wireless access. Examples are Bluetooth, WiFi and Remote-Keyless-Entry. Long distance wireless access is the last category. Interfaces in this part are GPS, digital Radio and mobile services. The authors use reverse-engineering and debugging to gain access to the on-board network in each category. For example through the CD player with a corrupt WMA file, though Bluetooth via stack vulnerabilities or addressable over mobile services. By introducing even more technologies for comfort and safety of cars the quantity of vulnerabilities will rise also. That is why a protocol like TSN has to be secure and robust to prevent the propagation of attacks over the in-car network architecture.

A specific attack is shown in the work of Koscher et al. [3]. They use the ODB-II diagnostic interface to attack the in-car systems. They extract all necessary data by analysing the output of individual ECUs. With this data extensive manipulations are possible. In the next step this manipulations are executed

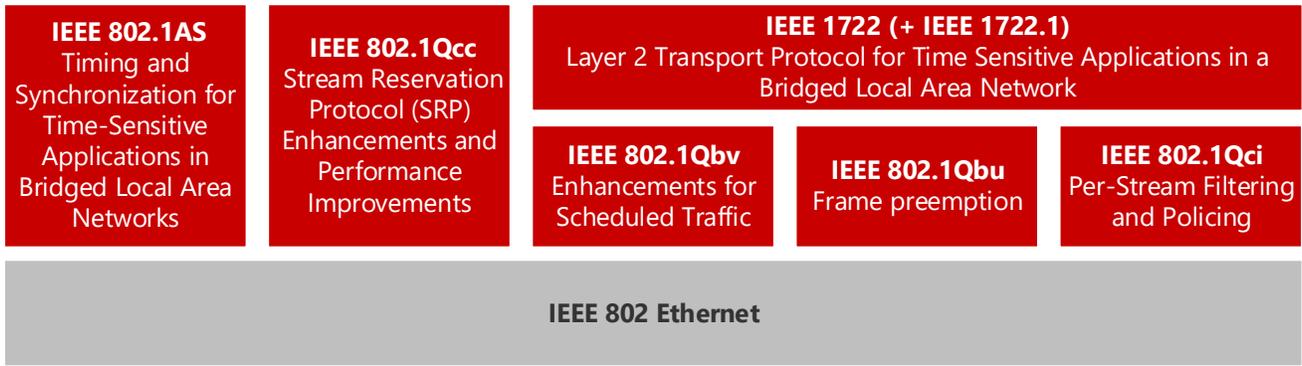


Fig. 1. Time Sensitive Networking Structure

through a PC connected with the OBD-II interface. This allows access to car components like radio, speed indicator, individual breaks and parts of the engine controls. A less specific but easier way to attack the in-car communication is a Denial of Service (DoS) attack on the busses which can prevent communication between ECUs. This is possible without the effort of analysing individual ECU output and could lead to the same disastrous outcome. In comparison this work examine the prevention of such DoS attacks in the next generation in-car communication. This should avoid corrupt ECUs from destroying the communication between others.

Henniger et al. [4] presenting a structured process to develop security requirements for in-vehicle networks. The base is a car with a heterogeneous bus system composed of CAN and FlexRay busses. The process includes four steps. First step is the identification of threats. This is enabled by attack trees. Here the root is a attack target and the leafs are sub goals that enable the superordinate target. All this targets and sub goals for the attacker are threats to the system. In the second step security requirements are identified which impede the identified threats. The last step is the prioritisation of the security requirements to determine the importance of each requirement. The risk is the indicator of the priority. It is calculated from occurrence probability, checkability and severity. The severity is a vector of different classes. So the damage to humans or financial loss can be rated individually. This process is the base for the discovery of threats to a TSN network in this work. The process is adapted to Ethernet in-car communication system.

Bouard et al. [5] present a middleware independent protection concept for IP based in-car communication. It provides concepts for encryption, authentication, policy administration and recognition of security violations. For the key administration secure hardware is required. Through modularity ECUs with different requirements can be implemented. In contrast this works focus is the transport layer. Weaknesses in this layer can undermine the security concepts in higher layers. Additionally the transition to a full IP based in-vehicle

network will be stepwise. So TSN have to support protection mechanisms.

III. BACKGROUND

This Section introduces the concepts of security, in-car networks and the TSN credit based shaping algorithm.

A. Security

The safety aspect is highly integrated in the automobile development process. This did not prevent unauthorised modification and stealing of sensitive information. Therefore security mechanisms have to be implemented in a system. A secure system is also a safe system but not reverse. A secure system provides [6]:

- Confidentiality: Secrecy of data objects. System information is only readable with an appropriate authorisation.
- Integrity: Prevention of unauthorised modification. Prevention of attacks with the target to modify the system behaviour.
- Availability: No loss in performance. Attacks can not corrupt the system performance.

The in-car networks of vehicles with advanced distributed technologies have to have this protection to prevent harm to automobile and passengers.

B. In-Car Networks

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

The future could be lead to a stepwise transition to Switched-Ethernet networks. Because of the higher bandwidth it is possible to transmit raw data streams of cameras, laser scanners and other sensors in those in-car networks. In addition the further development and acquisition is low priced through the wide distribution of Ethernet technologies. Higher layer protocols are easily adaptable as well.

To support the domain spanning communication of in-car functions a Switched-Ethernet network provides the technology to establish a structured architecture. The future target is to deploy one flat Ethernet network that provides the simultaneous transmission of messages with different priorities.

To maintain a safety communication so called real-time Ethernet protocols are used to guarantee quality of service. They extend the standard Switched-Ethernet protocol functionality.

C. Time Sensitive Networking

The Time Sensitive Networking [7] real-time Ethernet protocol is a collection of draft standards which are adapted to real-time control stream network requirements. Domains for this protocol suite are for example industrial control facilities or in-car networks. The figure 1 shows major parts of this collection.

The first one is the IEEE 802.1AS [8] for the time synchronisation of TSN applications.

Secondly IEEE 802.1Qcc [9] describes the dynamic route and bandwidth reservation over the network.

IEEE 802.1Qbv [10] provides forwarding and queuing descriptions for the simultaneous transmission of asynchronous and synchronous messages.

The next standard is IEEE 802.1Qbu [11]. It contains the function of frame preemption. This allows to interrupt a packet in transmission for a higher priority frame.

IEEE 802.1Qci [12] describes an ingress control through per-stream filtering and policing. This is where the later, in section IV, introduced Credit Based Metering takes place.

The Last one in this set of standards is the IEEE 1722 transport protocol specified in IEEE 1722 [13] and IEEE 1722.1 [14].

In figure 2 the path of a message through a TSN switch is shown.

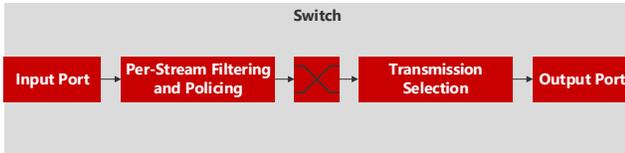


Fig. 2. Message Path in a TSN Switch

Firstly an incoming frame is forwarded to the "Per-Stream Filtering and Policing" block defined in IEEE 802.1Qci. The function of this is described in detail in the section IV-B.

The next step is the crossbar block which decides the route of the packet.

Each route guides to a "Transmission Selection" block and the connected output port. The selection implements the forwarding and queuing strategy of TSN IEEE 802.1Qbv.

Each TSN device output behaviour is regulated by the forwarding and queuing strategy. This is shown in detail in the Figure 3.

TSN packets could be assigned to one of eight different priorities (0 lowest - 7 highest priority). A switch has one

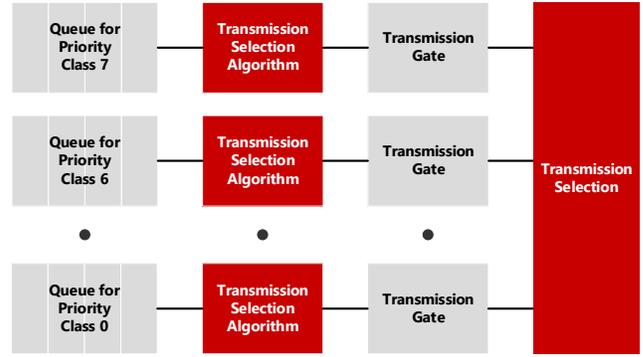


Fig. 3. IEEE 802.1Qbv Transmission Selection

queue per port and priority. Firstly a specific Transmission Selection Algorithm decides if a packet of its queue is ready to be send. One of this algorithms is later described in detail.

In the next step the gate has one of the two states "OPEN" or "CLOSE". This gate states can be static or change in a predefined timed behaviour. If the gate is in the state "CLOSE" the message has to wait. If its in the state "OPEN" the last step decides if it can be forwarded to the port.

Transmission Selection is a simple priority based decision. If more than one frame is ready to be send it chooses the one with the highest priority.

The important part in this work is Transmission Selection Algorithm. One of this algorithms is the Credit Based Shaping (CBS) standardized in the Audio Video Bridging (AVB) protocol [15]. It is the predecessor of TSN and specifies message streams that traverse the network over dynamic reserved routes. This reservation contains the maximum interval and size of such stream frames. Therefore the task of the CBS algorithm is to maintain this reserved bandwidth.

The dynamic stream reservation is called Stream Reservation Protocol (SRP) and is defined in the standard IEEE 802.1Qat [16].

The basic functionality is that a source (Talker) broadcasts information of a stream into the network (Talker Advertise). Each switch in this network saves this information and the destination of the Talker.

A sink (Listener) that wants to receive a stream answers with an acknowledge message (Listener Ready). Each switch on the route examine if enough bandwidth is free for the stream. If so the Listener Ready is forwarded in the destination direction and the switch reserved the bandwidth on the specific route. The next switch does the same till the message reaches the Talker.

Now the Talker starts to send the stream. On each device from Talker to Listener a CBS algorithm shapes the traffic flow to comply with the reserved bandwidth on the egress port.

The Credit Based Shaper is defined in IEEE 802.1Qav [17]. It is based on a credit value manipulated by two different gradients called "idleslope" and "sendslope". This gradients

are composed of the reserved bandwidth (RB) and the total bandwidth (B) of the connected egress port.

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

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

The CBS functionality is shown exemplary in figure 4.

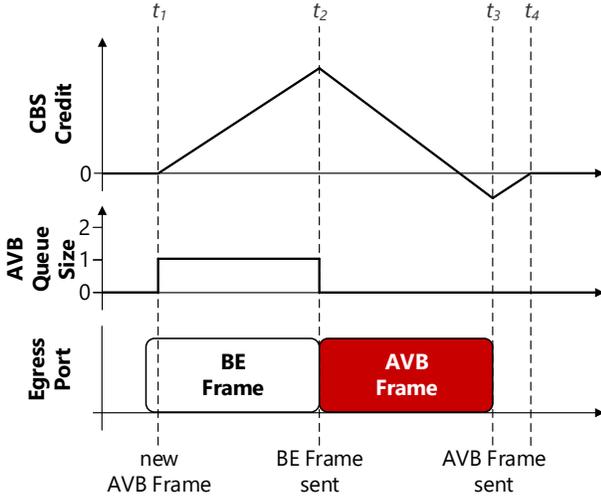


Fig. 4. IEEE 802.1Qav Credit Based Shaping example

At the beginning the credit start with 0. Whenever the credit is greater or equal to 0, an AVB frame is allowed to be transmitted. When no frame is sent and the credit is lower than 0, the credit increases according to the "idleslope" till 0 (see t_3 in figure 4). If the transmission of a frame is blocked by a higher priority frame, the port is occupied or the specific gate is in the state "CLOSE", the credit increases according to the "idleslope" above 0 (t_1 in figure 4). If an AVB frame is sent, the credit decreases according to the "sendslope" (t_2 in figure 4). If no frame is waiting in queue or in transmission the credit stays at 0 (t_4 in figure 4).

IV. PROTECTION THROUGH CREDIT BASED METERING

This Section introduces the attack scenario and the Credit Based Metering protection concept.

A. Attack Scenario

The attack scenario is build upon the previous presented protocols for stream reservation (SRP) and traffic shaping (CBS). The attack tree in figure 5 shows this attack scenario for an AVB switch.

The attack target is to delay or delete frames of a stream traversing the network. The trees root represents this target (see #0 in figure 5).

To enable this an attacker has to fill the queue with the class of the target stream to delay other frames or produce a overflow to provoke the switch to drop other frames (#1 in

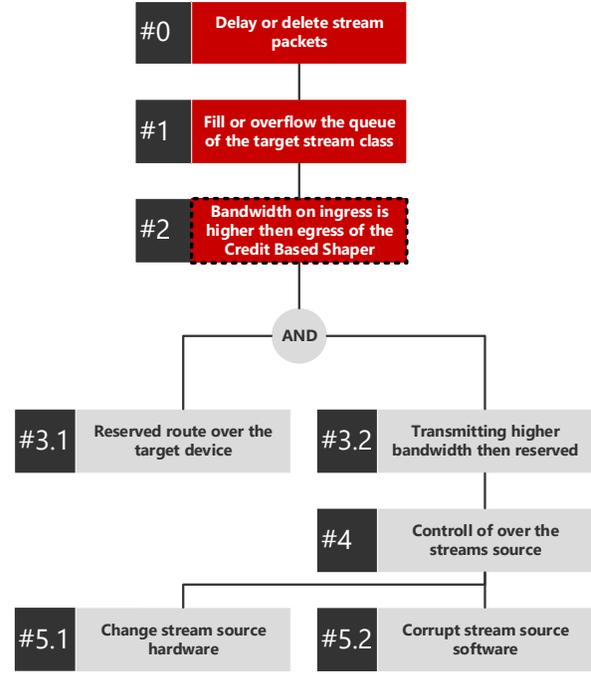


Fig. 5. DoS Attack Tree for an AVB switch

figure 5). This is possible when the input bandwidth is higher than the CBS is shaping on the output of a switch (#2 in figure 5).

Two criteria have to be fulfilled to produce a higher input bandwidth. First a reserved route over the target switch (#3.1 in figure 5) and secondly a device that sends a higher bandwidth than reserved (#3.2 in figure 5). By taking control over an already established stream source both criteria are achieved (#4 in figure 5). In other cases a Listener for the attack stream has to be connected behind the target switch.

To gain control over a Talker the attack has to corrupt the software or exchange the device running the software (#5.1 and #5.2 in figure 5).

Now the Talker under control can send frames in any pattern. All switches on the path to the Listener will try to forward this frames and queue them in the Transmission Selection queues. If the bandwidth of this transmission pattern exceeds the reserved bandwidth frames from other sources will be delayed or, if the queue is full, destroyed.

B. Preventing the Attack

To prevent those attacks for different traffic classes IEEE 802.1Qci is in the standardization process. Figure 6 shows the structure of this per-stream filtering and policing.

There are three levels a message has to pass through before it can be queued.

The first are the stream filters. They are configured to decide which gates and meters are responsible for handling a specific message stream id set.

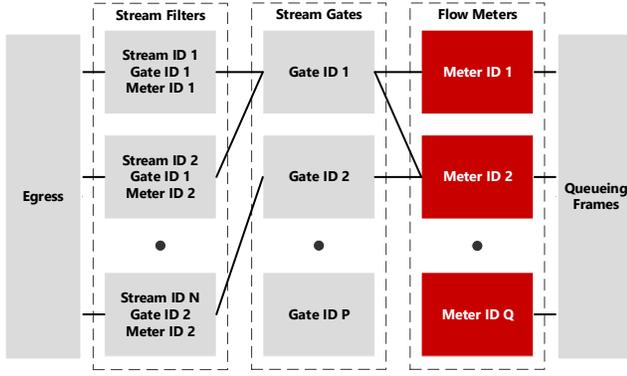


Fig. 6. IEEE 802.1Qci Per-stream filtering and policing

Secondly there are the stream gates. Those have one of the two states "OPEN" or "CLOSE". This state can change based on a predefined timed behaviour. If the responsible gate is in the state "OPEN" the message forwards to the defined meter. If the gate is in state "CLOSE" the message is dropped.

This flow meters contain individual algorithms to assert if a message is allowed to be queued. This work presents a concept for such a meter called Credit Based Meter. Like the name is hinting it is a specific ingress counterpart for the Credit Based Shaping egress behaviour.

In general the Credit Based Meter is based on a credit value like the CBS. But in this case the reception of the AVB like stream frames is only allowed when the credit is greater or equal to 0. When a stream frame is received and the credit is lower than 0 this frame will be discarded.

The Slopes "*idleslope*" and "*sendslope*" from equation 1 and 2 in section III-C are also used for the CBM. The only difference is that the reserved bandwidth (*RB*) is only the bandwidth of the stream the CBM is metering. Additionally the CBM contains a maximum burst size parameter ($Burst_{max}$) configuring the maximum count of stream frames that are allowed in an incoming burst. This is used in combination with the stream frame sending duration ($T_{duration}$) 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 divided by 1. So the definition of $Burst_{max} = 1$ results in $Credit_{max} = 0$.

The CBM state machine is shown in figure 7. It has two major states. They are "RUNNING RECEIVING ALLOWED" (R-RA) and "RUNNING RECEIVING FORBIDDEN" (R-RF).

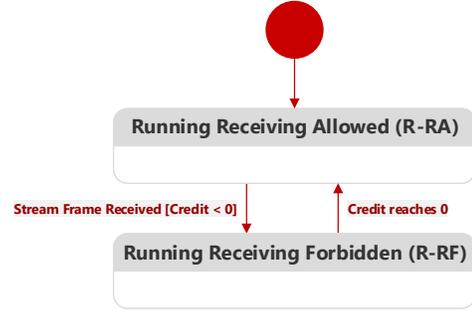


Fig. 7. Credit Based Metering state machine

When the CBM starts the state is R-RA. In this state the credit starts with 0 and increases according to *idleslope* till the first stream frame is incoming or the credit reaches the maximum. In the second setting the credit now stays at the maximum value until a stream frame is incoming.

In the R-RA state the credit is decreased by "*sendslope*" for the receiving duration of a stream frame. When the frame is received and the credit is greater or equal to 0 the credit increase again with "*idleslope*" or allows an additional stream frame reception and decreases again by "*sendslope*". When the credit reaches the maximum credit it stays on this value until a stream frame is incoming. If the credit is lower than 0 the state will be switched to R-RF.

In R-RF each incoming stream frame will be deleted. Simultaneously the credit increases with "*idleslope*". In the moment the credit reaches 0 the state is changed back to R-RA.

In figure 8 an example of the CBM algorithm behaviour is shown.

Firstly the state is R-RA and the credit is 0 and increases according to "*idleslope*" until the first stream frame arrives (see t_1 in figure 8). The credit decreases by "*sendslope*" for the duration of the stream frame (t_2 in figure 8). 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 stream frame arrives. This is delayed by an incoming best effort (BE) frame (t_3 in figure 8). The next stream frame arrives and the credit is decreased again (t_4 to t_5 in figure 8). The credit increases till the third stream frame receiving starts (t_6 in figure 8). And again the credit decreases by "*sendslope*" until the end of the frames duration.

More scenarios are presented later in the evaluation in section V-C.

V. EVALUATION

In this section describes the CBM evaluation. Therefore a credit simplification and a state chart model is proposed. And finally test scenarios are shown which are using the state chart model to evaluate the CBM algorithm.

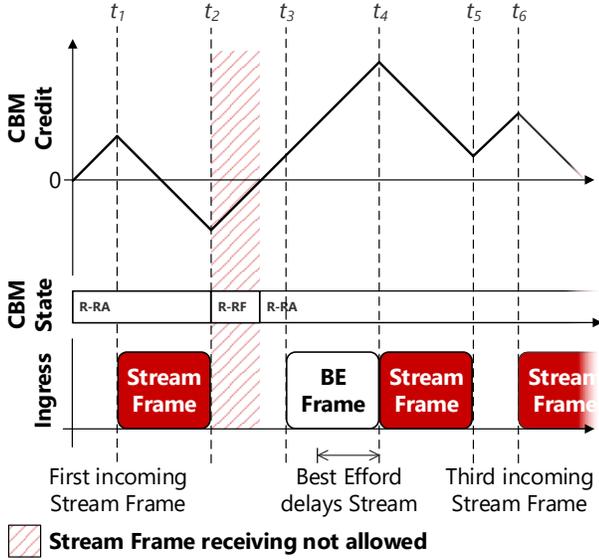


Fig. 8. Credit Based Metering example

A. Credit Simplification

To simplify the CBM model a credit simplification takes place. The target is to simplify the state model in a way that obtains all major characteristics to cover standard protocol and attack scenarios.

This is done by assuming that the reserved bandwidth (RB) in the test scenarios is 50% of the port bandwidth (B) in all cases. This results in stream frames with a max size of 50% of an Ethernet full size frame. The equations 5 and 6 showing the calculation of the "idleslope" and "sendslope" in this scenario with 100Mbit/s Ethernet ports.

$$\begin{aligned} idleslope &= RB \\ &= 50Mbit/s \end{aligned} \quad (5)$$

$$\begin{aligned} sendslope &= RB - B \\ &= 50Mbit/s - 100Mbit/s \\ &= -50Mbit/s \end{aligned} \quad (6)$$

Both slopes has the same value in positive and negative direction. Hence it can be assumed that each stream frame increases or decreases the credit by 1. Secondly to go on with the simplification all other not stream frames have just half or full size packets to result in a credit variance of 1 or 2. With this simplified credit behaviour the state chart model is designed in the next section V-B.

B. Credit Based Metering Model

The CBM state machine is designed with "Yakindu State-chart Tools" [18]. This tool allows the design, simulation and code generation of state machines. This Tool is used to design and evaluate the CBM algorithm.

Therefore the CBM state machine is designed in the first step. Each design iteration could be tested and validated directly with the simulation of the state machine. This is done until the final design is finished.

Now the designed CBM behaviour is verified. This is done with a CBS state machine generating traffic pattern for the CBM algorithm.

In the next step the generation of Java code is done. This code implements the CBM state machine so that scenarios can be implemented to algorithmic evaluate the CBM behaviour in boundary and attack conditions.

C. Test Scenarios

This section presents a selection of scenarios that are important boundary conditions to evaluate. They are evaluated with the implemented state machine. The selection contains protocol valid and attack scenarios.

1) *Valid Block by Time-Triggered Frame*: The first scenario describes the CBM behaviour in the situation that a valid stream frame is delay by a higher priority time-triggered (TT) frame. This arises because a stream frame is not send even if the port is not in use when this stream frame is too long to be send before the reserved TT window starts. This is configured through the timed gates of the IEEE 802.1Qbv Transmission Selection described in section III-C.

The CBM behaviour in this scenario is shown in figure 9.

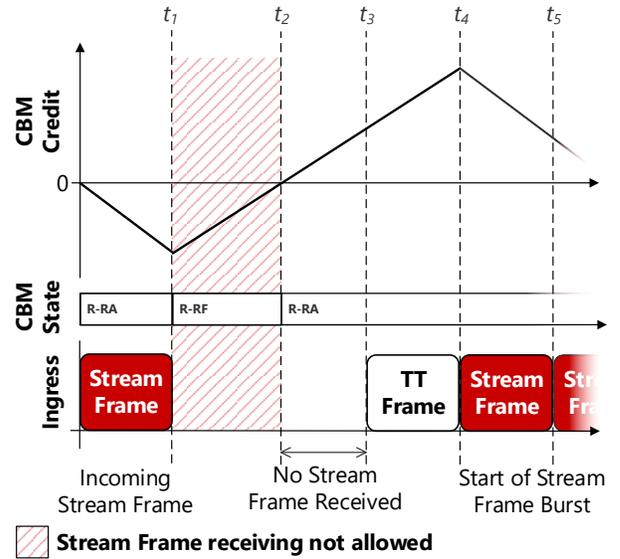


Fig. 9. CBM Scenario: Valid Block by Time-Triggered Frame

First an incoming stream frame results in a credit lower than 0 (see t_1 in figure 9). Now the credit increases back to 0 (t_2 in figure 9). At this time a new stream frame is expected but it is delayed by an upcoming TT window (t_2 to t_3 in figure 9). The CBM has no knowledge about this but increases as long as no stream frame is incoming or the maximum is reached. Next the TT frame is received and the credit behaves in the

same manner and increases according to "idleslope" (t_3 to t_4 in figure 9). Through the high credit value it is now allowed to receive a burst of stream frames like the CBS allows to send a burst (t_4 and t_5 in figure 9). Directly sequential receiving is allowed till the credit is lower than 0.

This scenario shows the CBS traffic behaviour that allows bursts to catch up the missed reserved bandwidth is allowed by the CBM algorithm.

2) *Attack with Simple Spam*: This second scenario shows how the CBM handles a simple spam attack by a corrupted source. It means that the sources traffic pattern is not protocol valid. It sends each stream frame directly after the one before and tries to DoS the target.

The reaction of the CBM to this traffic pattern is shown in figure 10.

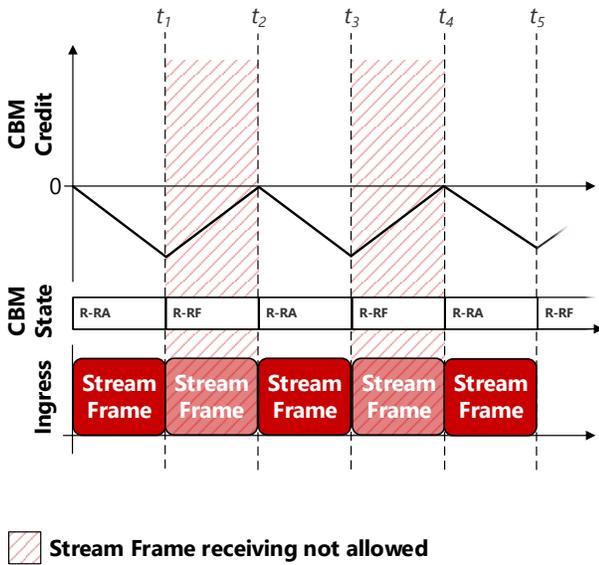


Fig. 10. CBM Scenario: Attack with Simple Spam

Because of the simplified credit the time for sending a stream frame and for the credit to reach 0 is the same. This results in the drop every other incoming stream frame (see t_1 and t_3 in figure 10). The frames that are incoming after the credit reaches 0 be kept (t_2 and t_4 in figure 10).

This shows that the kept frames bandwidth correlate directly with the reserved bandwidth of the stream.

3) *Valid reaching of Maximum Credit*: The third scenario present a valid worst case traffic pattern. The worst case in the simplified credit setting is the delay of a stream frame through a best effort frame blocking the port when the stream frame reaches the CBS and an additional time-triggered send window is to close to send the stream frame (SF) between both. This results in a burst of 4 stream frames by the CBS algorithm on the source.

Therefore the CBM maximum burst parameter ($Burst_{max}$) is set to 4. The equation 7 shows the calculation of the maximum credit within the simplified credit setting.

$$\begin{aligned} Credit_{max} &= sendslope * T_{duration} * (Burst_{max} - 1) \\ &= 1 * (4 - 1) \\ &= 3 \end{aligned} \quad (7)$$

According to the simplification in section V-A the combination of "sendslope" and " $T_{duration}$ " results in the credit increase 1. Thus the maximum credit in this scenario is 3.

In figure 11 the reaction of the CBM to this traffic scenario is shown.

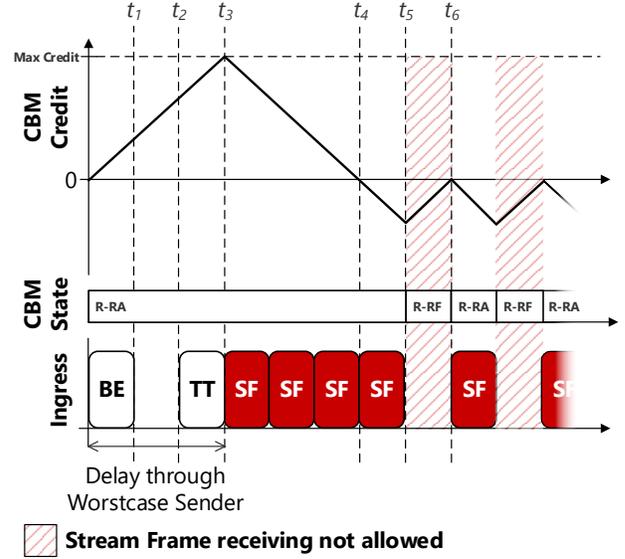


Fig. 11. CBM Scenario: Valid reaching of Maximum Credit

The scenario begins with an incoming BE frame. The credit starts at 0 and increases. After this the port stays free to not delay the following TT frame (see t_1 to t_2 in figure 11). With the receiving of the TT frame the credit reaches the maximum of 3 (t_3 in figure 11). Now the stream frame burst starts. Each incoming stream frame decreases the credit by 1 until the credit is 0 (t_4 in figure 11). At this time one additional stream is allowed to be received and the CBM switches to R-RF till the credit is back to 0. (t_5 to t_6 in figure 11). So a total burst of 4 stream frames are received.

This scenario points out that the maximum burst parameter has to be configured according to the expected worst case on the receiving port. Otherwise valid frames would be dropped.

4) *Reaching of Maximum Credit trough Attack*: In the fourth scenario a attacker with knowledge of the CBM tries to create a DoS on the target.

Instead of sending stream frames in a valid pattern the corrupted source waits to get the credit in the CBM as high as possible. When the credit is at maximum the source is bursting stream frames to the receiver.

Parameters are the same like in the previous scenario V-C3. This results in a maximum credit of 3 again.

Figure 12 illustrates the response of the CBM.

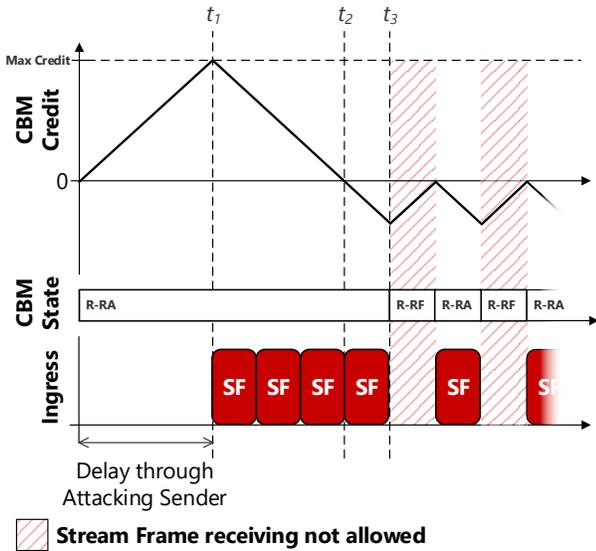


Fig. 12. CBM Scenario: Reaching of Maximum Credit trough Attack

Firstly the corrupted traffic source waits until the CBM credit reaches its maximum of 3 (see t_1 in figure 12). Now the stream frame burst is incoming and like before the last allowed incoming frame arrives when the credit reaches 0 (t_2 in figure 12). After the burst is finished just valid traffic patterns are allowed or the source has to wait again until the credit reaches the maximum to start an additional burst (t_3 in figure 12).

This reveals that the stream frame traffic pattern is the same as in the previous scenario V-C3 where the maximum credit is reached in a valid setting. When the source is alternating between burst and wait the result is that a not protocol valid traffic pattern is allowed by the CBM. But the overall bandwidth of the stream is still the same as the one reserved for this stream.

VI. CONCLUSION & OUTLOOK

Through the interconnected multitude of sensors and ECUs in today's vehicles the demand for a new communication technology lead the development to Ethernet. This seeds new vulnerabilities into the in-car network architecture. The CBM is a solution for a TSN meter algorithm or could be implemented additionally with the AVB protocol to protect the system against DoS attacks. It protects integrity and availability of an in-car network by individually controlling the stream input on each port in the network. As shown the CBM allows all valid traffic pattern of a CBS algorithm. To gain the best protection the maximum burst parameter has to be as low as possible. However it must allow the worst case scenario of a specific input port. This trade-off between severity and worst case estimation has to be considered.

In the future this CBM will be implemented in a network simulation environment to further analyse the best maximum burst configuration and the overall performance. Additionally

the mutual reaction with other TSN traffic shaper concepts will be evaluated in this simulation based analysis.

REFERENCES

- [1] P. Meyer, T. Steinbach, F. Korf, and T. 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*, Dec 2013, pp. 47–54.
- [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." *USENIX Security*, 2011. [Online]. Available: <http://www.autosec.org/pubs/cars-usenixsec2011.pdf>
- [3] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, and S. Savage, "Experimental security analysis of a modern automobile," in *2010 IEEE Symposium on Security and Privacy*, May 2010, pp. 447–462.
- [4] O. Henniger, L. Apville, A. Fuchs, Y. Roudier, A. Ruddle, B. Weyl, T. Paristech, C. Ltc, and S. Antipolis, "Security requirements for automotive on-board networks," pp. 641–646, 2009.
- [5] A. Bouard, B. Glas, A. Jentzsch, A. Kiening, T. Kittel, F. Stadler, and B. Weyl, "Driving automotive middleware towards a secure ip-based future," in *10th conference for Embedded Security in Cars (Escar'12)*, Berlin, Germany, November 2012. [Online]. Available: https://www.sec.in.tum.de/assets/staff/alexandre/Escar_Paper_final.pdf
- [6] C. Eckert, *IT-Sicherheit Konzepte - Verfahren - Protokolle*, 9th ed. Muenchen: Oldenbourg, 2014.
- [7] IEEE 802.1 TSN Task Group, "IEEE 802.1 Time-Sensitive Networking Task Group." [Online]. Available: <http://www.ieee802.org/1/pages/tsn.html>
- [8] —, "IEEE 802.1AS-Rev - Timing and Synchronization for Time-Sensitive Applications." [Online]. Available: <http://www.ieee802.org/1/pages/802.1AS-rev.html>
- [9] —, "IEEE 802.1Qcc - Stream Reservation Protocol (SRP) Enhancements and Performance Improvements." [Online]. Available: <http://www.ieee802.org/1/pages/802.1cc.html>
- [10] —, "IEEE 802.1Qbv - Enhancements for Scheduled Traffic." [Online]. Available: <http://www.ieee802.org/1/pages/802.1bv.html>
- [11] —, "IEEE 802.1Qbu - Frame Preemption." [Online]. Available: <http://www.ieee802.org/1/pages/802.1bu.html>
- [12] —, "IEEE 802.1Qci - Per-Stream Filtering and Policing." [Online]. Available: <http://www.ieee802.org/1/pages/802.1ci.html>
- [13] "Ieee standard for layer 2 transport protocol for time sensitive applications in a bridged local area network," *IEEE Std 1722-2011*, pp. 1–57, May 2011.
- [14] "Ieee standard for device discovery, connection management, and control protocol for IEEE 1722(tm) based devices," *IEEE Std 1722.1-2013*, pp. 1–366, Oct 2013.
- [15] Institute of Electrical and Electronics Engineers, "IEEE 802.1BA - IEEE Standard for Local and metropolitan area networks - Audio Video Bridging (AVB) Systems," IEEE, Standard IEEE 802.1BA-2011, Sep. 2011.
- [16] —, "IEEE 802.1Qat - IEEE Standard for Local and metropolitan area networks - Virtual Bridged Local Area Networks - Amendment 14: Stream Reservation Protocol (SRP)," IEEE, Standard IEEE 802.1Qat-2010, Sep. 2010.
- [17] —, "IEEE 802.1Qav - IEEE Standard for Local and metropolitan area networks - Virtual Bridged Local Area Networks - Amendment 12: Forwarding and Queuing Enhancements for Time-Sensitive Streams," IEEE, Standard IEEE 802.1Qav-2009, Dec. 2009.
- [18] itemis AG, "YAKINDU Statechart Tools." [Online]. Available: <https://www.itemis.com/en/yakindu/state-machine/>