Scalable Automotive Intrusion Detection Systems

From the ECU to the Vehicle Operation Centre

Dr. Eduard Metzker, Dr. Maximilian Engelsberger, Vector Informatik GmbH, Stuttgart

Abstract

Many new vehicle functions are based on the intensive meshing of on board and off board services. This introduces the challenge to protect these vehicle functions against cybersecurity threats. Besides other security controls the automotive industry is introducing intrusion detection systems (IDS) to mitigate these threats. In this article we present a scalable wholistic solution that spans from the ECU to the vehicle operation center (VSOC).

Cybersecurity as an Enabler

Cybersecurity is an enabler for new vehicle features which are realized via a mesh of on board and off board functions. The automotive industry has successfully introduced several security controls for protecting these vehicle features against threats. Examples for such security controls include signed software update containers, secure boot, and secure onboard communication.

IDS as an Additional Automotive Security Control

Motivated by new standardization [1] and regulation efforts [2] Intrusion Detection Systems (IDS) are gaining attention. The following figure gives an overview of the detection and mitigation process of cybersecurity events to which IDS contribute.

1 Security related events are detected and recorded on the ECUs of the vehicle. The data is filtered and aggregated on board.

2 The aggregated data is transmitted to the security operation center (SOC).

3 In the backend the data is refined and enriched to provide maximal value for the security analysts.
4 The fleet data is analyzed by the security experts of the SOC to determine the root cause and impact.
5 Based on the nature of the incident threat responses are developed. This could include the development and test of a software update.
6 The software update is rolled out to affected vehicles of the fleet to fix the vulnerability.

In the rest of this article, we focus on the steps 1-4 of the overall process.

**Fig. 1: IDS as part of the security incident detection and mitigation process**

Although IDS are a well-known security control in classical IT systems, they are not yet widely established in the automotive industry. This is currently changing and a scalable, distributed IDS architecture was standardized by the AUTOSAR consortium [3].

**Standardization of On-Board IDS solutions**

The 20/11 release of the AUTOSAR specifications [4] provide a solution for several IDS challenges:

1 **Clearly defined components and interfaces:**
IDS according to AUTOSAR consist of clearly defined components: Security event sensors detect security events (SEv) which are indicators of a potential security incident. The security
event sensors report the SEv to the local intrusion detection system manager (IdsM). The IdsM buffers and filters the reported SEv. If SEv pass the filters of the IdsM they are regarded as qualified security events (QSEv). QSEv can be passed to the local security event memory (Sem) for storage and/or they can be passed to an ECU with a backend link. The intrusion detection system reporter (IdsR) on this ECU collects all QSEv of the vehicle. The IdsR can optionally enrich the received QSEv before propagating them to the VSOC for evaluation by security analysts.

2 Standardized protocol for transmission of security events

The AUTOSAR IDS protocol allows for the efficient transmission of QSEv to other ECUs. It facilitates interoperability of IDS components from different vendors which can communicate over different bus systems.

3 Standardized Security Sensors and Events

All AUTOSAR based systems share the same base layer architecture. So, several security events which could be detected by the base layer modules can and have been standardized. This allows OEMs to focus on the security events and sensors which are specific to their products while at the same time they can use the standardized sensors and events as a baseline to start from.

4 Scalability

The approach provides the required scalability for the deployment of on-board IDS in different vehicle architectures. A starting point could be a single ECU IDS which records the QSEv in its local secure diagnostic memory. This could be expanded with an IdsR which receives the QSEv and transmits them to the SOC. The deployment could be further extended to other security related ECUs which also have IDS requirements. The following figure illustrates the deployment of a distributed on-board IDS based on the described components.
Fig. 2: Example for the deployment of IDS components in a vehicle architecture

Fleet to VSOC Backbone

The fleet to VSOC backbone is a subsystem of the overall Intrusion Detection System (IDS). It consists of in-vehicle and backend-aligned components which both realize different tasks of data transfer and processing on their way from the ECUs into the vehicle security operation center (VSOC). The fundamental problem is to transfer huge amounts of security events which are generated by dozens of ECUs as fast as possible and as complete as needed into the VSOC over unreliable mobile networks. Therefore, a reduction of the amount of data from Gigabits per second and vehicle (Gbit/(s*vehicle)) down to a few Kilobytes per burst and vehicle (KB/(burst*vehicle)) or some Megabytes per vehicle and day (MB/(day*vehicle)) is performed by upstream filter chains, see the Intrusion Detection System Manager (IdsM). The fleet-to-VSOC backbone has three primary responsibilities:

1. Translation regarding transfer technologies and data semantics
2. Refinement of security events with value-added data
3. Preservation of the customer’s data ownership

The subsequent description focuses on the application-specific aspects. The common system design, including scalability and provisioning of the backend services are not in scope of this article.
System Component Overview

The fleet-to-VSOC backbone consists of an in-vehicle component named IDS Reporter (IdsR) and a backend component named IDS Listener (IdsL). The IdsR receives standardized security events from the individual ECUs, buffers them temporarily and transmits them to the IdsL. The immediate data transmission depends on the availability of the mobile network connection. The IdsL is the counterpart to the IdsR and the logical link between the vehicle fleet and downstream VSOC applications and other data sinks. Besides of the reception of the incoming security events from the fleet, it acts as a semantic translator between the automotive domain and the IT domain. While in the automotive domain the AUTOSAR standard is of high importance, the IT domain comes along with standards and protocols like Syslog and/or custom IDS data formats used in Security Information and Event Management (SIEM) systems. Not only a field-by-field mapping is necessary, but rather a semantic translation and explanation of the automotive-specific data. Finally, the IdsL helps the OEM to preserve full control over its data ownership and to provide well-defined handover points to third-party applications. These can be, for example, the OEM's archive databases, real-time dashboards, or third-party analysis tools like the previously mentioned SIEM systems.
Stakeholders and Roles

A simplified constellation for the described system consists of the stakeholders and roles as listed in the following table:

Table 1: Stakeholders and Roles of the fleet to VSOC backbone

<table>
<thead>
<tr>
<th>Stakeholder</th>
<th>Localization</th>
<th>Description of the role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle manufacturer or vehicle fleet operator</td>
<td>Vehicle, Backend</td>
<td>Creates and provides security-relevant data to the manufacturer or operator of in-vehicle and backend-aligned IDS components. Receives security incident reports from the manufacturer or operator of a SIEM system.</td>
</tr>
<tr>
<td>Manufacturer or operator of in-vehicle and backend-aligned IDS components</td>
<td>Vehicle, Backend</td>
<td>Collects, transmits, and refines security-relevant data from the vehicle manufacturer or vehicle fleet operator and provides it to the manufacturer or operator of a SIEM system.</td>
</tr>
<tr>
<td>Manufacturer or operator of a SIEM system</td>
<td>Backend</td>
<td>Receives security-relevant data from the manufacturers or operator of in-vehicle and backend-based IDS components. Analyses and visualizes those data and provides incident reports to the vehicle manufacturer or vehicle fleet operator.</td>
</tr>
</tbody>
</table>

Event Processing Pipeline

Inside the backend-aligned IdsL component, a highly customizable event processing pipeline may be configured. It offers the possibility to perform various stages of customer-specific data refinement. Thereby, two fundamental principles are applied for all processing stages:

1. **Preservation of raw data**: No data shall be dropped implicitly in order not to harm any downstream analysis
2. **Principle of value-added data**: Annotation of raw data with labels or metadata with the goal to generate added value to downstream analysis or visualization
The event processing pipeline is used to semantically process the raw data coming from the vehicle and make them easier to interpret for downstream analysis tools and security analysts. Further processing steps comprise customizable validity checks, the aggregation of security events with third-party data sources, as well as to ensure compliance with customer-specific data protection and data privacy policies. Further examples are given in greater detail in the subsequent sections. The functionality of each processing stage can be implemented either system-internally or -externally. The decision of the localization of an actual processing stage depends on whether a suitable system-external processing stage already exists or whether further domain knowledge is required for its implementation. Each product-internal processing stage may use both product-internal and -external data sources via specific interfaces. The decision of the localization of the data source depends on whether a suitable product-external data source already exists or not.

**Data Sources**

The primary functionality of the data source is to receive Qualified Security Events (QSEv) from the IdsR components in the individual vehicles or a fleet of vehicles via an Over-the-Air (OTA) communication interface. An important aspect of this interface is to support the different requirements of the vehicle manufacturers and vehicle fleet operators regarding their communication channel between an individual vehicle and the backend. By now, a data source is supported which uses Vector vConnect for the OTA communication.

**Data Processing Stages**

Various processing stages can be customized for the individual needs and linked into the Event Processing Pipeline in sequential, parallel, or mixed order. As stated above, the primary goals are to preserve raw data and to add valuable information to the datasets for downstream analysis. Subsequently, a list of possible processing types and examples is given:

- **Semantic Data Refinement**: The principal goal is to increase the readability and interpretability for subsequent analysis tools as well as for their users, which may be a security analyst. A first example might be the lookup with a registry of QSEv, which adds a human-readable name (e.g., “Ethernet MAC collision”) and a description (e.g. “Receive local MAC address as source of an external frame”) instead of forwarding just the Event-ID 87. Those
metadata are part of the AUTOSAR standardization. Similar examples might be the lookup with a registry of vehicle-specific information like ECU details and details about their integration into the overall E/E architecture of the vehicle.

- **Data Validity:** This category of data processing stages checks incoming QSEv for its validity and plausibility. Depending on the result, the dataset is labeled as valid or invalid QSEv or can – additionally – be annotated with valid values or value ranges for certain fields. Again, this gives the security analyst a better understanding and a better possibility for evaluation. An example could be a check whether the IdsM header information is valid. Therefore, a check might be performed if the received ECUIDs are built into the given vehicle or not. If the ECUID was faked, this might be a reason for having a closer look.

- **Data Aggregation:** This category aggregates and correlates relevant data from third-party systems with the given QSEv or group of QSEv which might have a temporal or spatial proximity to the received datasets. This might come from data sources within the same vehicle or from the vehicle’s environment. Thereby, the data sources of high interest are exactly those, which are not accessible or not easily interpretable by the downstream analysis tools. Examples of such third-party data sources might be data collector tools in the vehicle, system log and trace data or data from the in-vehicle resource management like CPU and memory load. Another group of examples might be decision data produced by Autonomous Driving Assistance System (ADAS) or even data from traffic lights in close proximity to the vehicle at the time when the QSEv was generated.

- **Data Protection:** Not to be confused with Data Security, this category is about all measures to prevent situations of data loss and providing traceability of security incidents. That is why it might be necessary to customize a long-term data storage of all incoming QSEv of the fleet. Having a long-term archive of recorded security data might be necessary to enforce regulative policies as well. As mentioned above, most of the data coming from the vehicle need some form of explanations with describing annotations. As the QSEv raw data comes along with a very special structure, encoding and data content, it is not meaningful to forward all of this to the downstream SIEM. A possible solution could be to dump all incoming raw data into the archive for a certain period while only refined data elements are forwarded to the SIEM.

- **Data Security:** Not to be confused with Data Protection, this category is about all measures to prevent situations in which collected data is disclosed to third parties without the
explicit intention and permission to do so. To ensure this, it is necessary to customize processing steps of anonymization or pseudonymization of QSEv elements. An example might be to remove personal-related data from the QSEv before they are forwarded to a SIEM to fulfill regional data protection regulation policies. Relevant questions might be a) Where the data will be transferred to: Which country, etc., b) What regulations are to be applied there and c) How must the given data be processed to fit into those regulations. At this stage raw data might get removed from the actual dataset, but if this stage is meaningfully combined with an archiving stage (see point Data Protection) no raw data will be lost.

As a QSEv has passed all stages of a given Event Processing Pipeline it was annotated with descriptive metadata or annotation labels. Following the postulated principle of preserving the raw data, those annotated QSEv are now called Extended Qualified Security Events (XQSEv).

**Data Forwarding**

The primary functionality of the given component is to forward XQSEv produced by the Event Processing Pipeline to different kinds of data sinks. Those might be system-internal or external data sinks. To give only a few examples, those might be third-party analysis tools like SIEM systems or customer-specific databases for raw data recording. Another category might be web-frontends which work as dashboards and allow to visualize live data streams in a control center.

**Summary and next steps**

In this article we described a wholistic approach for Vehicle IDS based on the AUTOSAR IDS standards. The approach covers a crucial part of the overall cybersecurity incident detection and mitigation process which is requested by regulators. Based on the standard components, interfaces and protocols, a distributed vehicle IDS can be deployed in a wide range of vehicle architectures. Furthermore we see a big potential in coupling the IDS capabilities with automotive network firewalls. In this scenario the network firewall acts as a security sensor for the IDS while also protecting the communication at the same time by dropping suspicious traffic. This creates an effective and modular intrusion detection and prevention system (IDPS). Considerable added value is introduced by the data enrichment and refinement enabled by the IdsL component. We see a big potential if the data stream from the vehicle fleet is enriched by...
the IdsL with data from the vehicle development process and other sources. After this automotive domain specific preparation of the data is finished enterprise-grade of the shelf SIEM and SOC software can used to act on the data. This reduces the cost of the overall solution and prevents the creation of information silos.


