Toward Safety and Security Development by Identifying Interfaces of Automotive Functions

Toru Sakon and Yukikazu Nakamoto
Graduate School of Applied Informatics
University of Hyogo
Kobe, Japan
Email: tr.sakon@gmail.com, nakamoto@ai.u-hyogo.ac.jp

***Abstract*—An item is a system or an array of systems used to implement a function at the vehicular level. However, owing to increasing demands for advanced functions and security features in an automobile, an item and the manner in which it is defined has become more complex. In this research, we propose using resource sharing as the basis for defining an item and its boundary. We use four simple categories for introducing an influencer that represents a shared resource and its management function. This makes the process of defining an item simple and straightforward. Further, by refining an influencer, complex interaction between systems in an item is better described. We applied this method to define an item for the sample systems.**

Keywords–ISO 26262; item definition; interface; resource sharing; management; security.

I. INTRODUCTION

Safety in case of malfunction or failure of a vehicular system is a priority in every vehicle. To ensure safety, the functional safety approach is adopted in the development of an electrical and/or electronic (E/E) system in a vehicle. ISO 26262 is a widely adopted international standard for ensuring functional safety of an E/E system in a vehicle [1]. In ISO 26262, the development targets are termed as items. Items are defined by their functions at the vehicular level. Definition of an item boundary is essential for an item. However, in the current advanced vehicular functions, there are some cases in which an item boundary cannot be clearly defined. The functions at the vehicular level are realized using more than one in-vehicle Electronic Control Unit (ECU). However, in the case of an ECU shared by multiple items, interference between the items may result via shared resources of the ECU. For advanced vehicular level functions, multiple items are closely integrated to realize the function. The development of cybersecurity measures is indispensable to the development of safety functions, considering the cyberattacks on the safety functions. In this context, during the development of security functions on the basis of items, the characteristics of attack points are necessary for defining items. However, security attack points for the items are not necessarily included in the items defined by in-vehicle function and their boundaries. In other words, for the development of secure advanced functions in vehicles, we need a sophisticated method to formalize all interactions among the target items. Reflecting this fact, compositional aspects are added to the item definition in the latest draft standard [2]. However, an interaction based model of a combined system requires tightly-coupled processes calling each other's interfaces. This increases the complexity in designing combined items. In this study, we introduce a new object called an influencer to maintain conventional item definitions. An influencer defines shared resources and management functions for the items. With the introduction of an influencer, we propose a method to incorporate the newly introduced high-functional impacts and security considerations as an interface requirement into the item definition while maintaining the granularity of a conventional item definition.

The structure of this paper is organized as follows. Section II briefly describes the difficulties in designing combined items for automotive safety and security. We state the basic idea of an influencer and use cases of the designing procedure with an influencer, in Section III and IV, respectively. Section V describes related works. We present conclusion in Section VI.

II. DEVELOPMENT OF FUNCTIONAL SAFETY IN AN AUTOMOTIVE E/E SYSTEM

The functional safety design of an E/E system is central to safety design of an automobile. Functional safety means "the absence of risks due to hazards caused by the malfunctioning behavior of E/E systems" [3]. ISO 26262 is a functional safety standard for automotive electronic systems. Its development process is defined based on a V-shaped process. Item definition starts in the early stage of the development process (See Fig. 1).

An item is defined as "a system or array of systems to implement a function at the vehicular level, to which ISO 26262 is applied" [3]. The boundary of an item with that of other items is an item's important property. It is defined considering a) elements of the item; b) environment of the item; c) interactions of the item with other items or elements; d) functionality required by other items; e) functionality required

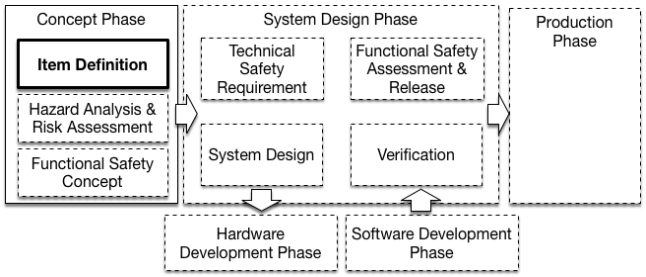

Figure 1. Development process of ISO 26262 and the target of this study (bold line)

 ISBN: 978-1-61208-643-9

from other items; f) allocation and distribution of functions; and g) operating scenarios [1].

However, in general, defining a highly complex system by different subsystems and determining the boundary and interaction between them is a difficult task. In addition to the complexity of the system itself, the complexity of mutual communication between the elements of the complex system is very high. Therefore, the structure of interactions between the subsystems is highly dependent on the way, in which a complex system is subdivided. Further, it is difficult to eliminate all the hidden communication paths in a complex system. An automobile is a highly complex system and hence, it is difficult to define an item and its boundaries during its development. We categorize these difficulties into three types. The first category is the definition and description of boundaries related to potential sharing, particularly non-message-like associations that do not appear explicitly. For example, consider an item consisting of one or more systems. Furthermore, another item shares one or more of its systems. In this case, there is an interaction between two items via shared systems. A typical example is sharing an ECU with multiple items. In this case, there are some mutual effects owing to sharing of resources, such as memory, Central Processing Unit (CPU) usage, Input / Output (IO), or basic software resources. However, sharing of ECU resources is not expressed explicitly. Therefore, it is difficult to identify items to incorporate item definition.

The second problem is defining sophisticated item boundaries during the differential development of complicated functions. A simple example is given in [4] but for advanced vehicle functions, multiple items cooperate to realize the function. For example, a Lane Keeping Assist System (LKAS) and a Parking Assist System (PAS) are realized by integrating several vehicular level functions, such as a steering, throttle control, braking, and the functions that control them. From the aspect of reusing a proven system, it is preferable for a developer of differential development to minimize and isolate the side-effects of the new function from the preexisting product. Furthermore, for the development based on item definitions, items should be isolated by their boundaries as clearly as possible. To satisfy both these objectives, one of the rational ways is to handle predefined items as shared resources, define the management of item, and finally introduce the newly defined shared resource and its management function.

The third problem is developing an item definition method considering the factor of cybersecurity. One of the serious threats to car cybersecurity is the scenario where the vehicle safety function is compromised by cyberattacks. A malicious attacker must search the attack path to the function first, in order to compromise the vehicle safety function. If the attacker finds the path, he can begin the attack on the function. In ISO 26262, the safety functions of vehicles are studied on an item and its boundary basis. The attack path to the safety functions should be through the item boundary. Therefore, it is reasonable to subject an item and its boundary to an analysis of car cybersecurity. This means that, in threat analysis, influencers are candidates for attack surfaces for items. The entry point of a threat is assumed to be located at an influencer. Threat analysis method will be applied to the item and related influencer for the assessment of the threat. For cyberattack against items, there are direct attacks on functions defined in items and indirect attacks, such as attack on resources used by functions. In indirect attacks, as per the current item definition in ISO 26262, the management is possibly not considered as an item because resource management is not directly related to vehicular level functions. As a result, “feature” is introduced as a subject of security instead of an item in [5]. Furthermore, there are cyberattacks from the attack point that they are not included in the item. For example, in an in-vehicle network, a cyberattack may be possible from a compromised ECU, which is not included in the item. Therefore, there is a need for a method to explicitly describe cybersecurity requirements in the item of functional safety development.

III. SOLUTION TO THE PROBLEMS

The problem categories exist due to the difficulties of considering the indirect communication between items. The first and third category of problems arise from not considering indirect communication via ECU hardware, basic software, or architecture. The second category of problems arises from difficulties in describing the complex communication patterns of advanced functions between the items which are based on mutual communication between systems belonging to an existing product. In other words, these problems are due to the interface being examined from the aspect of direct interaction. An example of relationship that cannot be found only from the decomposition of functions is of two independent systems sharing a Controller Area Network (CAN) network. These two systems have no interfaces to each other. However, once one of the systems is compromised and starts a Denial of Service Flooding attack on the network, the other system is also affected. In order to realize a function which consists of multiple elements, we propose a method to define the item and its boundary based on the interaction between the shared resources and their management. By explicitly describing this shared structure, we aim to organize and add potential relationships between items as functional additions and resources that can be added to items and reflect them in the development after item definition. This newly introduced structure is called an influencer. First, resource sharing among items is classified by resources while their management method by four simple categories, i.e. categories of influencers. The types of resources are logical resources (information) and physical resources (hardware, physical resources etc.). Resource management includes transferring and sharing of resources (TABLE I). Each cell of TABLE I is a definition of an influencer between items. TABLE I lists four categories of influencers.

- Movement of information resources corresponds to data movement between items. The information may consist of a command, message, or other data. An example of a management mechanism is a communication protocol.
- Sharing information resources is equivalent to referring the same data from multiple items. Examples of management functions of shared memory are exclusive control, distributed shared memory management protocol, and operating system resource control.
- Movement of physical resources is accompanied by movement of physical objects, for example, electric

 ISBN: 978-1-61208-643-9

power transfer during charging. An example of management mechanism is power delivery control.

- Sharing of physical resources implies sharing media and resources, such as memory, communication media, buses, and power supply. The management methods include bus arbitration and time division control.

In this paper, the boundaries of items are defined as follows:

1) Define shared resources. If prerequisite architecture is available, it may be used to define shared resources.
2) Define the way of management of shared resource to fit into one of the categories listed in TABLE I. This is the initial description of the influencer.
3) Refine the initial description of the influencer to assign it to each item. In this procedure, the influencer may be divided into sub-influencers and define the interaction between them.
4) Allocate the defined (sub) influencer to each item. In other words, the influencer is defined as an element that describes sharing and management of information, data, physical quantities, and their mechanisms among the items.

IV. USE CASES

In this section, we provide some use cases.

1) In case of the definition and description of boundaries related to potential sharing, we must identify their hidden interaction. We suppose the implementation of two items on a single ECU as an example (Fig. 2). Simply sharing an ECU i.e. hardware, implies sharing its hardware resources, such as CPU, available memory, and IO systems (Fig. 2 a). Next, we define the management of these resources. In most ECUs, the CPU is managed using a time division approach and memory is managed by a preassigned fix region and a dynamically allocated region. The shared IO between two items is managed in an exclusive manner (Fig. 2 b). Thus, we have three candidates for the influencer. We now focus on the IO system. The IO resource is managed exclusively; for an exclusive control, we need a protocol between resource requester (in this case, an item) and provider (in this case, the influencer). At this point, we can refine the influencer as the combination of two influencers. One is the management of the IO itself and the other is accessing data and its management protocol (Fig. 2 c).

2) In case of the item definition and its boundaries in differential development, we need to define them to minimize the modification from predefined items. In this study, we assume a PAS consisting of a steering system, brake system, throttle system, and a PAS controller system (Fig. 3). We assume that the first three systems are already well defined as preexisting products and they consist of a user interface system and a control system for the control target. In other words, each item of a preexisting product consists of user interface and controller functions (Fig. 3 a). In this case, the user interface or controller functions share control of the steering, brake, and throttle system. Then, we define status data of a PAS shared among the items and management of its exclusive control. These are the resources and controls required for the initial description of the influencer (Fig. 3 b). Next, for refinement of the influencer, the initial influencer is decomposed into two parts namely, the shared status and consistency control of it. These parts are assigned to each item of the predefined product and the newly defined PAS controller system (Fig. 3 c).

TABLE I. CATEGORIES OF INFLUENCERS

	Transferred	Shared
Logical resources (Information)	Communication (Message transfer, Remote Procedure Call)	Data Sharing (Shard memory, Shared object) or Code (Function)
Physical resources	Physical Transfer (Battery Charging)	Physical Sharing (Communication bus, Battery)

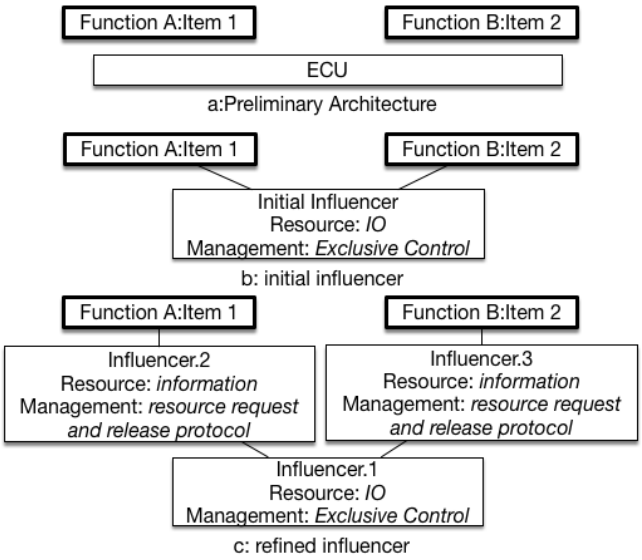

Figure 2. An ECU shared by items

3) In case of the item definition method considering cybersecurity, we need to define an item and its boundary to identify potential cyberattacks to it. We consider some items connected by a network as an example (Fig. 4). In this case, we assume a CAN network (Fig. 4 a). The network itself is physically shared. Further, the data frames on the network are logically shared and managed following the priority and arbitration rule. In other words, this is the initial definition of the influencer (Fig. 4 b). A malicious attacker can attack a target item by compromising another item connected to the network. To take this attack into consideration, we decompose the initial influencer into two categories. One is the network influencer, whose resources are the network and bus arbitration. The other is the node influencer that sends and receives information from the network influencer. In this scheme, an item and its boundary have the node influencer as their boundary. A malicious attack can be formulated using the malicious data received from the network. Thus, by adding the node influencer for item definition, we can take cyberattack from other item as receiving malicious data on node influencer (Fig. 4 c). Moreover, changing the management of a network influencer, the attack condition on a node influencer may be changed. For example, dividing a network into two sub-networks and

 ISBN: 978-1-61208-643-9

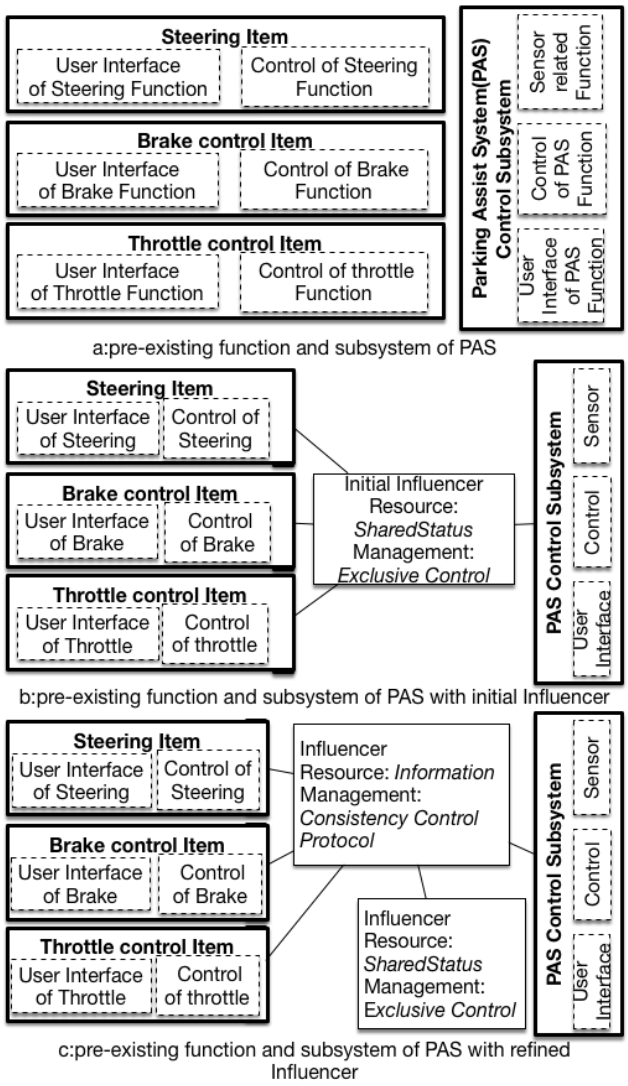

Figure 3. Advanced complex item

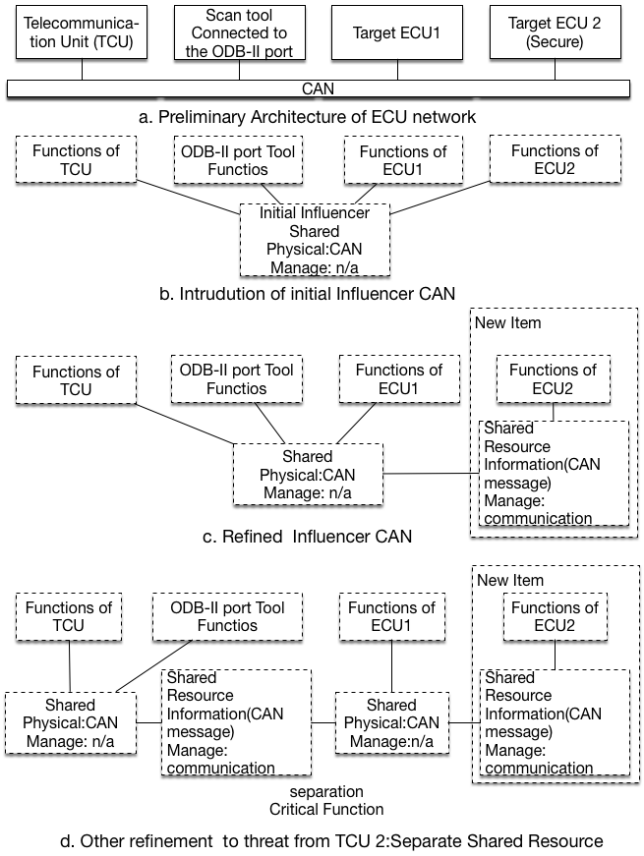

Figure 4. Item definition of a secure in-vehicle system

connecting them by a new item such as a gateway ECU, attacks from the other sub-network may be restricted (Fig. 4 d).

V. Related works

There are considerable previous studies on the functional decomposition of the system as well as studies that perform hierarchical decomposition on the functional basis for vehicles [6]. However, it is necessary to assume anomalies and attacks from parts that do not directly have a functional relationship from the system complexity and response to cyberattacks in the future. Such a relationship is not included in the relationship between functions. In this research, we infer that the method of functional decomposition focused on the sharing side is more comprehensive than that of based on the relation between the functions.

VI. Conclusion and remarks

In this study, we proposed a new item definition method by introducing an influencer. Particularly, focusing on logical and physical resources shared by item boundaries and their management, we proposed a method to extract the construction elements of influencers using four simple categories. We formulate that by sharing resources, remote cyberattacks are also captured by the granularity size of conventional items. In the definition of sophisticated functions that consolidate multiple items, the proposed method facilitates decomposing items. In the future, we would like to examine the effectiveness and designing technique of the multi-layered defense of an in-vehicle system with an influencer as the key.

Acknowledgment

This work is partly supported by JSPS KAKENHI Grant Numbers 16H02800 and 17K00105.

References

[1] ISO, Road vehicles - Functional safety - Part 3: Concept phase , ISO 26262-3:2011 (E).

[2] ISO, Road vehicles - Functional safety - Part 3: Concept phase , ISO/FDIS 26262-3:2018 (E).

[3] ISO, Road vehicles - Functional safety - Part 1: Vocabulary, ISO 26262-1:2011 (E).

[4] ISO, Road vehicles - Functional safety - Part 10: Guideline on ISO 26262 ISO 26262-10:2012 (E).

[5] SAE International. Cybersecurity guidebook for cyber-physical vehicle systems, SAE J3061 201601.

[6] W. Hoxk, C. Witteveen and M. Wooldridge, Decomposing constraint systems: equivalences and computational properties, Proc. 10th Int. Conf. of Autonomous Agents and Multiagent Systems, 2011. pp.149-156.

 ISBN: 978-1-61208-643-9