Advanced Science and Technology Letters
Vol.139 (SecTech 2016), pp.333-339
http://dx.doi.org/10.14257/astl.2016.139.68

Threat Modeling for Automotive Security Analysis

Zhendong Ma and Christoph Schmittner

AIT Austrian Institute of Technology, Digital Safety & Security Department,
Donau-City-Strasse 1, 1220 Vienna, Austria
{Zhendong.Ma, Christoph.Schmittner.fl}@ait.ac.at

Abstract. Connected and intelligent vehicles create new risks to cybersecurity and road safety. Threat modeling is a building block in automotive security engineering that identifies potential threats for corresponding mitigations. In this paper, we address how to conduct threat modeling for automotive security analysis during the development lifecycle. We propose a practical and efficient approach to threat modeling, extending existing tool support and demonstrating its applicability and feasibility.

Keywords: security, automotive, threat modeling, safety

1 Introduction

Cars are becoming more and more intelligent and connected. On the flip side, this technological transformation also makes modern vehicles vulnerable to cyberattacks [1, 2]. Cars used to be closed systems. The automotive systems were not designed with security in mind. Recent security breaches in the automotive domain raise the issue in the industry and the public, making it clear that security is a critical concern with an impact on public and road safety, especially when new technologies such as autonomous driving and intelligent transport systems (ITS) are becoming reality.

Rigorous security engineering to the development of automotive systems is required to address safety and security of modern vehicles. Security analysis is one of the important building blocks in this process. Threat modeling is a technique for security analysis. As a concept, threat modeling has been extensively covered in many previous works. However, as we observed, there are many misconceptions and confusions on how to apply threat modeling in an efficient and correct way, especially in the emerging field of automotive security. In this paper, we provide a practical guide on conducing threat modeling for automotive system security analysis. Moreover, we propose optimizations to make it more efficient, repeatable and accurate. We also show that our proposal is readily supported by existing tools for practical need in the automotive industry.

In the following, Sec. 2 gives an overview of secure development in the automotive domain. Sec. 3 describes our approach to threat modeling, followed by a proof-of-concept in Sec. 4. Sec. 5 concludes the paper.

ISSN: 2287-1233 ASTL

2 Secure Development of Automotive Systems

The automotive industry traditionally has a very high quality and safety standard. As a basis, the automotive industry developed and accepted ISO 26262 [3] as the standard for generic road vehicle functional safety for electrical and electronic (E/E) systems that cover both hardware and software. The development starts with the *concept phase* in which an item is defined followed by activities such as hazard analysis and risk assessment (HARA) and the definition of functional safety concept. An item is a system or an array of systems to implement a function at the vehicle level to which ISO 26262 is applied. In the next phase *product development*, the functional safety concept is refined to produce technical safety requirements and hardware and software system are designed, integrated, and tested. Compliance and correctness of the safety goals and their implementation are validated. Safety cases are produced as evidence for compliance and certification. As security becomes an issue for safety in modern vehicles, there are on-going discussions on how to seamlessly integrate security activities into existing safety-oriented automotive development lifecycle [4]. The SAE J3061 standard [5] is the most prominent in the industry to define secure development process for cyber-physical vehicle systems. It builds on ISO 26262 and intends to compliment the safety process with security process with interaction points between the two engineering processes. It defines Threat Analysis and Risk Assessment (TARA) to identify potential cybersecurity threats, assess and rate the risk associated with the threats. Threat modeling is specified in J3061 to identify threats and security risks during design. In the literature, *Macher et al.* [6] proposed to extend HARA with threat modeling STRIDE method for security-aware hazard analysis and risk assessment (SAHARA) to define the ASILs, i.e. add STRIDE-based security analysis as an additional activity to the safety analysis of items defined according to ISO 26262. *Eichler et al.* [7] proposed a modular and flexible approach for security risk assessment in the automotive development process.

3 Automotive Threat Modeling

Threat modeling *per se* is the activity of defining a theoretical model of perceived threats to a system. The better the assumptions, the closer is the theoretical model to the practical implementation to capture the significant attack vectors [8]. Therefore, threat modeling can be seen as addressing two basic questions: 1) How to model a system and its trust assumptions? 2) How to model an adversary that captures its motivations, capabilities, and actions and its tactics, techniques, and procedures?

Fig. 1 shows the conceptual view of applying threat modeling technique to automotive security analysis. Solid and dotted arrows indicate information flows. Threat modeling as an activity should be performed in all phases (concept, product development, and production and operation) of the lifecycle. Although the basic technique remains the same, threat modeling will have different input with respect to the details of the system as it evolves along the development lifecycle. Moreover, threat modeling will have different objectives in each phase. In the concept phase, threat modeling is based on system concept and high-level system design with less technical details. The outcomes are high-level security requirements and the security

concept. In the product development phase, the input to the threat modeling will be system design specifications as well as implementation details. The objective of threat modeling is to define technical security requirements for functional and security design, discover design vulnerability and flaws, and specify comprehensive security requirements that can be verified and validated in unit and integration testing along the V model. It is very likely that threat modeling will be an iterative process due to the continuous development and modification of system design and implementation details. In the production and operation phase, threat modeling serves as a preparation for conducting actual penetration testing on finished automotive components and systems. It identifies high-risk inputs and keeps a checklist of things to audit which helps to prioritize entry points that could yield the most return during a pentest. Since threat modeling includes the definition of not only threats but also mitigations, outcomes from threat modeling might have significant impact and modification to the design and implementation of the automotive systems in the development lifecycle.

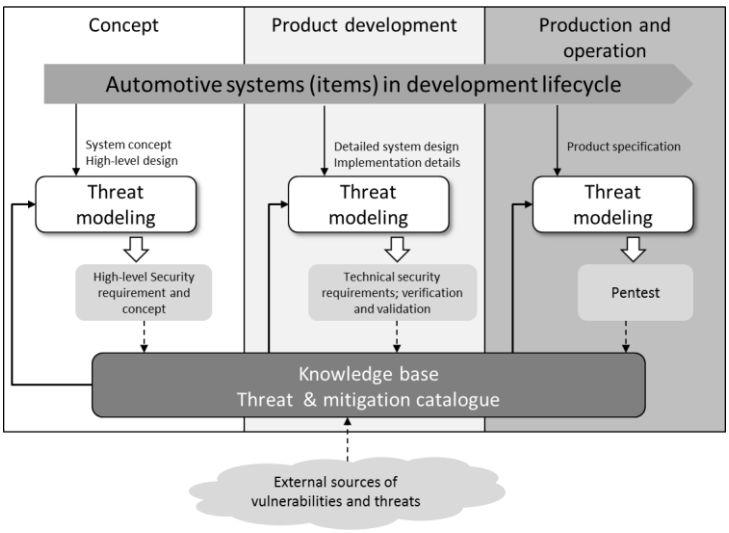


Fig. 1. Overview of systematic threat modeling in automotive secure development lifecycle

The knowledge base is a central store of information on threats and the corresponding mitigations, which is used in threat modeling in different phases concerning various system models. The knowledge base should be continuously enriched by the output from threat modeling activities with additional threats and mitigations, enabling the reuse of threat modeling artefacts throughout different projects. Further, related vulnerabilities and threats from external sources such as vulnerability databases, hacker communities, and security researchers should be timely incorporated into the threat and mitigation catalogue. The dotted arrows indicate that ingress information to the knowledge base which requires processing to match the format and semantics of the threat and mitigation catalogue. Although human expertise will always play a main role in this process, the establishment and maintenance of a knowledge base in which threats and mitigations are collected, categorized, and updated that are applicable to the context of different system

diagrams will be a viable way to increase efficiency and reduce cost and human errors. In such a way, complex system can be analyzed semi-automatically by leveraging previous results; repeated work can be kept at minimal. This also allows reusing analysis efforts for future projects and even across domains. Knowledge databases for web security can be used as an example for considering threats to the backend and web-communication parts of an automotive update system. As we will show in the next section, current tool is able to support such a vision.

The main focus of threat modeling is software. Generally, it models a system in Data-flow Diagram (DFD). There are five types of elements in a DFD diagram: process, data store, data low, external interactor, and trust boundary. A process can be any software component that takes input and performs actions and/or generates output. Processes can have different levels of granularity. A high-level process can be decomposed into low-level processes in a hierarchical way, e.g. a Level 0 process "Head Unit" can be decomposed into Level 1 processes of "Communication Gateway", "Linux OS", "Applications", and "HMI" etc. Depending on the available system details and threat identification needs, a process can be further decomposed into lower-level components such as specific Linux kernel modules. Example data stores can be firmware, filesystem, or memory. A data flow represents the flow of data between elements, e.g. a data flow can be a protocol specific communication link such as CAN Bus, FlexRay, or HTTPs. An external interactor is either a human user or a user agent that interacts with a process from the outside. Trust boundaries divide the elements in the diagram into different trust zones, e.g. elements reside in the in-car systems and external hosts communicated from untrusted open networks.

Based on the DFD, one can identify threats stemmed from data flows by using a threat identification methodology such as STRIDE and assess the severity of the threats. The Spoofing, Tampering, Repudiation, Information disclosure, Denial of Service, and Elevation of privilege (STRDE) methodology is one of the most popular ones partially due its easy-for-developer origin and extensive documentation of applications [9]. However, depending on the granularity of the system information available and the timing of the threat modeling in the development lifecycle, alternative methodologies can also be used for optimal cost-benefit results.

Mitigations are technical or organizational countermeasures to the threats. The linking of mitigations to the threats ensures that all identified threats will be considered and addressed. Moreover, it also puts mitigations into perspective with the overall security architecture as well as other requirements such as usability, safety, and budget constraints for making sound design decisions. Validating theoretical models against actual systems will ensure the correctness of the results from the threat modeling. Validating all identified threats are addressed provides additional layer of quality control on the security process.

4 Implementation

In the past years, Microsoft has developed a tool called Threat Modeling Tool (TMT) [10]. The latest release in 2016 also provides possibilities to create new threat templates, which enables us to extend TMT so that it is suitable for threat modeling for automotive systems. When creating a new template, the most important parts are

stencils for drawing DFD diagram and *threat types* that define threat and mitigation catalogues. We create stencils for automotive components such as Electronic Control Unit (ECU), in which we add additional details to describe the component. Once defined, these additional details provide rich information about an automotive component during threat modeling. The threat catalogue can be defined by threat properties in the TMT template. Each threat type includes title, threat description. More importantly, it has fields of *include* and *exclude*, which can be used for writing simple logical expressions such as `source is [stencil name]` so that threats can be automatically generated on an element of a DFD diagram when the condition is satisfied. Threat properties are grouped by threat types. For example, in the default TMT template, the threat types are defined as spoofing, tampering, repudiation, information disclosure, denial of service, and elevation of privilege. Similar to stencils, threat properties can be flexibly extended for addition fields. Accordingly, specific mitigations can be defined in the corresponding threat properties.

Fig. 2 shows the proof-of-concept implementation of our threat modeling for automotive security analysis. It is based on the TARA of an automotive cockpit unit for remote access functionalities including maintenance and over-the-air (OTA) software update [11]. In the center of the figure is the Operator controller which is an ARM-based System-on-Chip (SoC) microcontroller running on embedded Linux. The Human-machine Interface (HMI) enables an operator in the cockpit to issue command and monitor the status of the vehicle. It communicates wirelessly with update severs at the back-end. Operators and engineers can access the controller remotely through a VNC client. The controller has a firmware data store within the physical boundary of the vehicle and it also connects to ECUs through the CAN bus interface. The right panel right shows customized stencils with system-specific details.

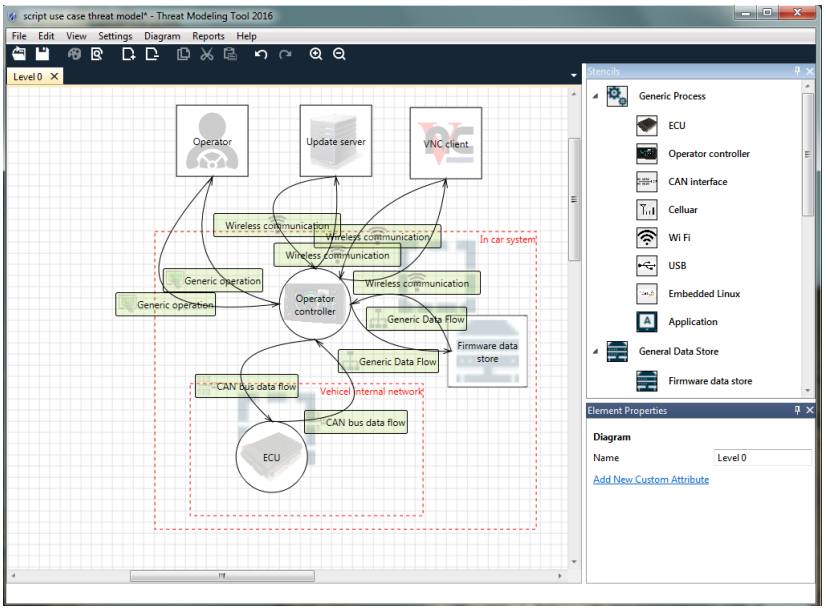
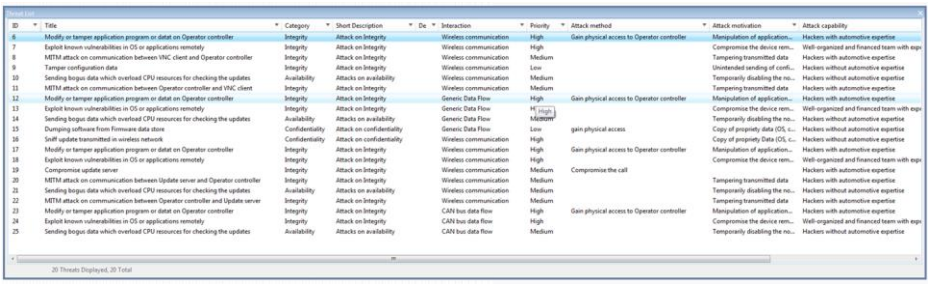


Fig.2. Example of top level DFD diagram of automotive unit

Fig. 3 shows the automatically generated threats based on the DFD diagram in Fig. 2. Due to space constraints, mitigations are not shown. Using a generic CIA method, we enumerate and identify the attacks on confidentiality, integrity, and available.



ID	Title	Category	Short Description	De	Interaction	Priority	Attack method	Attack motivation	Attack capability
6	Modify or tamper application program or datat on Operator controller	Integrity	Attack on Integrity		Wireless communication	High	Gain physical access to Operator controller	Manipulation of application...	Hackers with automotive expertise
7	Exploit known vulnerabilities in OS or applications remotely	Integrity	Attack on Integrity		Wireless communication	High		Compromise the device rem...	Well-organized and financed team with expe
8	MITM attack on communication between VNC client and Operator controller	Integrity	Attack on Integrity		Wireless communication	Medium		Tampering transmitted data	Hackers with automotive expertise
9	Tamper configuration data	Integrity	Attack on Integrity		Wireless communication	Low		Unintended sending of confi...	Hackers without automotive expertise
10	Sending bogus data which overload CPU resources for checking the updates	Availability	Attacks on availability		Wireless communication	Medium		Temporarily disabling the no...	Hackers without automotive expertise
11	MITM attack on communication between Operator controller and VNC client	Integrity	Attack on Integrity		Wireless communication	Medium		Tampering transmitted data	Hackers with automotive expertise
12	Modify or tamper application program or datat on Operator controller	Integrity	Attack on Integrity		Generic Data Flow	High	Gain physical access to Operator controller	Manipulation of application...	Hackers with automotive expertise
13	Exploit known vulnerabilities in OS or applications remotely	Integrity	Attack on Integrity		Generic Data Flow	H... High		Compromise the device rem...	Well-organized and financed team with expe
14	Sending bogus data which overload CPU resources for checking the updates	Availability	Attacks on availability		Generic Data Flow	Medium		Temporarily disabling the no...	Hackers without automotive expertise
15	Dumping software from Firmware data store	Confidentiality	Attack on confidentiality		Generic Data Flow	Low	gain physical access	Copy of propriety data (OS, c...	Hackers without automotive expertise
16	Sniff update transmitted in wireless network	Confidentiality	Attack on confidentiality		Wireless communication	High		Copy of propriety Data (OS, c...	Hackers without automotive expertise
17	Modify or tamper application program or datat on Operator controller	Integrity	Attack on Integrity		Wireless communication	High	Gain physical access to Operator controller	Manipulation of application...	Hackers with automotive expertise
18	Exploit known vulnerabilities in OS or applications remotely	Integrity	Attack on Integrity		Wireless communication	High		Compromise the device rem...	Well-organized and financed team with expe
19	Compromise update server	Integrity	Attack on Integrity		Wireless communication	Medium	Compromise the call		Hackers with automotive expertise
20	MITM attack on communication between Update server and Operator controller	Integrity	Attack on Integrity		Wireless communication	Medium		Tampering transmitted data	Hackers with automotive expertise
21	Sending bogus data which overload CPU resources for checking the updates	Availability	Attacks on availability		Wireless communication	Medium		Temporarily disabling the no...	Hackers without automotive expertise
22	MITM attack on communication between Operator controller and Update server	Integrity	Attack on Integrity		Wireless communication	Medium		Tampering transmitted data	Hackers with automotive expertise
23	Modify or tamper application program or datat on Operator controller	Integrity	Attack on Integrity		CAN bus data flow	High	Gain physical access to Operator controller	Manipulation of application...	Hackers with automotive expertise
24	Exploit known vulnerabilities in OS or applications remotely	Integrity	Attack on Integrity		CAN bus data flow	High		Compromise the device rem...	Well-organized and financed team with expe
25	Sending bogus data which overload CPU resources for checking the updates	Availability	Attacks on availability		CAN bus data flow	Medium		Temporarily disabling the no...	Hackers without automotive expertise

Fig. 4. Automatically generated threats

5 Conclusion

Security is one of the biggest challenges to connected and intelligent vehicle. Threat modeling is an effective technique to identify threats and mitigations during security analysis of automotive systems. We demonstrated that threat modeling, using existing tools, can be a useful and efficient analysis method for automotive security in different phases in the automotive development lifecycle.

Acknowledgments. This work is partially supported by the EU ARTEMIS project EMC2 (contract no. 621429) and Austrian Research Promotion Agency (FFG).

References

1. Checkoway, S. *et al.*: Comprehensive Experimental Analyses of Automotive Attack Surfaces. USENIX Security Symposium. (2011)
2. Miller, C., Valasek, C.: Remote Exploitation of an Unaltered Passenger Vehicle. Technical Report (2015)
3. International Organization for Standardization: ISO 26262 Road vehicles - Functional safety (2011)
4. Schoitsch *et al.*: The Need for Safety & Cyber-Security Co-engineering and Standardization for Highly Automated Automotive Vehicles. AMAA 2015, Berlin, Germany (2015)
5. SAE International: J3061 Cybersecurity Guidebook for Cyber-Physical Vehicle Systems (2016)
6. Macher, G., Sporer, H., Berlach, R., Armengaud, E., Kreiner, C.: SAHARA: A security-aware hazard and risk analysis method. 2015 Design, Automation & Test in Europe Conference & Exhibition (DATE), Grenoble (2015)
7. Eichler, J., Angermeier, D.: Modular risk assessment for the development of secure automotive systems. Conference: 31. VDI/VW-Gemeinschaftstagung Automotive Security, At Wolfsburg, Volume: VDI-Berichte 2263 (2015)

8. Dykstra, J.: Essential Cybersecurity Science - Build, Test, and Evaluate Secure Systems. O'Reilly (2015)
9. Shostack, A.: Threat Modeling: Designing for Security. John Wiley & Sons (2014)
10. Microsoft Threat Modeling Tool 2016
11. https://www.microsoft.com/en-us/download/details.aspx?id=49168
12. Schmittner, C., Ma, Z., Reyes, C., Dillinger, O., Puschner, P.: Using SAE J3061 for Automotive Security Requirement Engineering. SAFECOMP Workshops (2016)

