Attack Modeling for System Security Analysis
(position paper)

Abdullah Altawairqi and Manuel Maarek

Heriot-Watt University, Edinburgh, UK
{aha37,M.Maarek}@hw.ac.uk

Abstract. Approaches to the safety analysis of software-intensive systems are being adapted to also provide security assurance. Extensions have been proposed to reflect the specific nature of security analysis by introducing intention as a causal factor to reaching unsafe state of the system, or by introducing new layers in the system modelling to model its surface of attack.
In this paper we propose to extend these approaches by modelling the attacks perspective alongside the system. We explain how such modelling could be used to verify the coverage of the security analysis and facilitate its maintenance.

Keywords: hazard analysis, security analysis, attack model

1 Introduction

The ubiquitous computing paradigm is prevalent in nowadays systems which leaves them vulnerable to faults [5]. Safety analysis is essential for safety-critical systems as it identifies the behaviour and properties that each component and the system as a whole need to satisfy. The security of computer systems is of growing concern and requires such analysis to consider external threat as well as hazard in system design, development and operation. The combination of system safety and security using advanced engineering techniques along with detailed knowledge of domains and processes has made the task extremely challenging for cyber-security professionals.

In this paper, we explore security analysis approaches inspired by safety hazard analysis methods. We then present an extension to these approaches to model the attack perspectives alongside the system and discuss how this modelling could increase the security analysis by focusing on the attacker's point of view, and could help in verifying the coverage of the security analysis and facilitate the maintenance of the analysis.

2 Safety and Security Analysis

In this section, we explore a selection of works that adapt, for security, hazard analysis methodologies developed for safety. In particular, we are looking at

the base on which the analysis is built, the manner individual elements of the analysis differ, and how the coverage of the analysis is sought with the perspective of constructing a safety or security assurance case. A summary is presented in Table 1 of Section 2.3.

These safety analysis methodologies are Functional Hazard Analysis (FHA) and Systems-Theoretic Process Analysis (STPA) [6], part of Systems-Theoretic Accident Modeling and Processes (STAMP) [5]. For a more exhaustive survey of approaches combining safety and security analysis, see [4].

2.1 Safety Analysis

FHA. Safety analysis implies a combined systematic inspection of the system's functional specification and the conditions that could trigger hazards the system should avoid. FHA suggests a methodology to conduct such analysis by putting together a bottom-up and a to-down approaches. The bottom-up approach, Failure Mode and Effects Analysis (FMEA), exhaustively explores the ways each component of the system could fail, and the cause and safety implication of the failure. At the base of this elicitation of single point of failures are the system's components and specification. It is complemented by the top-down approach, Fault Tree Analysis (FTA), which structures the analysis of more complex scenarios that could lead to a hazard. Hazards are at the root of this tree-based analysis which explores the conditions which could trigger the hazard to occur, the conditions could be a combination of failures and events.

STPA. Due to the ever increasing complexity of systems, such systematic exploration starting from the system's architecture or from the hazards to avoid is costly to implement and maintain, and does not handle well systems with complex interactions. As part of STAMP [5] which is a more holistic approach to modelling accident causality, the hazard analysis STPA [6] was proposed as an alternative method based on a system abstraction in terms of controller and actuators. This approach allows to better model the system-wide interactions. STPA shifts the safety focus from failure to control. It regards hazard as resulting from a lack of enforcement of safety constraints rather than resulting from from component failure. This focus is more relevant for software-intensive systems as software components do not fail the same way physical component fail. The goal is to control the behaviour of the components and systems as a whole to ensure that safety constraints are enforced in the operational system.

Safety constraints are enumerated by mapping hazards to the system's control actions. Accident scenarios are then derived from these constraints by looking at control factors and control flaws.

2.2 Safety Analysis Methods Adapted for Security Analysis

STAMP and STPA were presented as system-centric analysis, [10, 11, 1] suggests with STPA-Sec that this approach could be applied for security analysis simply

by adapting its safety terminology to the security equivalent. STPA-Sec proposes a change to the traditional bottom-up approach of security analysis where threats are used to derive security requirements. A number of works [8, 2] have then highlighted the limitations of this safety-oriented view and proposed ways to effectively extend STPA to match the peculiarities of security analysis. In the following we explore how safety approaches have been extended to allow for security analysis.

Threat Model Based on Intention. In [8], the authors suggest that security threats could not simply be viewed as equivalent to a list of safety hazards but that the intention, which is at the heart of a security attack, must be modelled. This follows previous work in [7] by the authors to adapt the FMEA safety method for security by including vulnerabilities, threat agent and threat mode in the failure elicitation. They named their approach Failure Mode, Vulnerabilities and Effects Analysis (FMVEA). For a security-critical analysis, four ingredients (vulnerabilities, threat agent, threat mode, threat effect) are proposed and from which an attack probability is derived.

Similarly, an approach to extend FTA with attacker's intention has been proposed in [9]. Security events are added to the fault tree with a likelihood level. Note that this probability level is to change over time depending on availability of attack capabilities.

Surface of Attack. In [2], STPA-SafeSec is proposed as another extension of STPA for combined safety and security analysis. The authors suggest that the security analysis needs to be performed on the components of the system rather than the controllers as it is done in STPA. This choices allows to base the security constraints on the system physical vulnerabilities, effectively modeling the surface of attack of the system. Methods similar to STPA are then used to derive the security constraints and attack scenarios. In STPA-SafeSec, both safety constraints and security constraints are derived.

2.3 Base, Elements and Coverage of Analysis

Table 1 summaries the analysis strategies we discussed in this paper in terms of their strategy and coverage. The coverage should be seen as central for building a security assurance case from the analysis.

3 Attack Capabilities for Security Analysis

We outline in this section our proposition to extend the approaches discussed in Section 2. This extension has three main features: adding dependencies to the attack surface, modelling threats from the attacker's perspective, verifying the coherence between attack models and attack dependencies. We finally enumerate the changes to the STPA-SafeSec analysis process these features imply.

Table 1. Strategy and assurance of analysis methodologies
This table gives an overview of the strategy each analysis methodology from Section 2 implies and the assurance it provides in terms of its coverage.

Approach	Safety	Security	Base for the analysis	Differentiation criteria for analysis elements	How coverage of the analysis could be achieved
FMEA	×		Components	Cause and effect	Enumeration of failure modes
FMVEA		×	Components	Vulnerability, threat, attack type	Enumeration of threat modes
FTA	×		Hazards	Accident scenario	Decomposition of hazard causes
STPA	×		Control actions	Accident scenario (control factors and flaws)	Enumeration of four types of unsafe control action
STPA-Sec	×	×	Control actions	Hazard scenario	Enumeration of four types of unsafe control action
STPA-SafeSec		×	Components	Hazard scenario	Enumeration of attack modes per security concern
	×		Control actions	Hazard scenario	Enumeration of unsafe control action

3.1 Extend Attack Surface with Vulnerability Dependencies

STPA-SafeSec introduces on top of STPA's control layer a component layer. Each type of component of this layer is paired with generic security constraints (or vulnerabilities) to effectively map the physical surface of attack of the system.

This surface of attack is an essential base for the security analysis, we propose to extend it with vulnerability dependencies to identify the combinations of vulnerabilities that may result in the system being compromised. A dependency between a vulnerability v_1 and a vulnerability v_2 indicates that if v_1 is exploited it makes v_2 more open to subsequent exploits. These dependencies are expressed at the component and control layer of the system to refine the physical and control interactions of the system.

3.2 Model Attack alongside System

FMVEA and the extension of STPA-Sec suggested in [8] propose to replace safety-oriented failures with security-oriented threats to represent intentions in a security analysis.

We propose to go one step further in representing the intention by modelling attackers and attacks alongside the modelling of the system. An attack agent represents an attacker entity, and an attack mode represent an individual attack a method to exploit a vulnerability (these are respectively equivelent to FMVEA's threat agent and threat mode). An attack gain represents the outcome of an attack from the agent's point of view, this differs from FMVEA's threat effect which is expressed in terms similar to a failure effect in safety and therefore is from the system perspective. Finally, an attack strategy is represented as an attack tree [3]. An attack strategy is attached to an agent with an attack gain objective. Individual attack scenarios within a strategy are combined with *or* nodes and comprise attack modes exploiting components which are combined with with *and* nodes. Each individual attack scenario is evaluated by means of attack impact which is its hazard effect on the system.

3.3 Verification and Coverage of Analysis

The separation between the system and the attacks' point of view make it possible to perform some automated verification of coherence and coverage of the analysis. The correspondence between attack modes and vulnerabilities could help to validate the vulnerability dependencies and the attack strategies as Table 2 explains. Dependencies and strategies should strengthen the maintainability of the analysis by highlighting the impact a new vulnerability or attack mode.

Table 2. Correspondence, verification, coverage between system and attack elements

System	Mapping and Analysis	Attack
Vulnerability	Mapping indicating that an attack vector could exploit a given vulnerability	Attack mode
Vulnerability dependency	Individual attack scenario make use of a combination of attack modes on components, these combinations must correspond to vulnerability dependencies, every vulnerability dependency must be illustrated in attack scenarios	Attack strategy

3.4 Analysis Process

The extensions we suggest requires additional steps within the STPA-SafeSec [2] analysis process. These additional steps are indicated in Table 3.

4 Conclusion

We proposed in this paper to model attacks alongside the system to better capture the intention of the attacker and the attack vectors when deriving scenarios. This approach which extends previous works offers opportunities to verify the analysis and its coverage.

Table 3. Extension to STPA-SafeSec's analysis process
Note that we use the term vulnerability in place of STPA-SafeSec's security constraint

STPA-SafeSec process	Additional analysis step
High level analysis	1 Derive vulnerabilities 2 Define attack modes 3 Map attack modes to vulnerabilities 4 Define attack profiles (agents, gains)
Control loop analysis	1 Define vulnerability dependencies 2 Identify attack scenarios 3 Verify coherence between attack scenarios and vulnerability dependencies 4 Evaluate individual attack scenario by means of its effect, gain and modes

References

1. Abdulkhaleq, A., Wagner, S., Leveson, N.: A Comprehensive Safety Engineering Approach for Software-Intensive Systems Based on STPA. Procedia Engineering 128, 2–11 (2015)
2. Friedberg, I., McLaughlin, K., Smith, P., Laverty, D., Sezer, S.: STPA-SafeSec: Safety and security analysis for cyber-physical systems. Journal of Information Security and Applications (2016)
3. Jhawar, R., Kordy, B., Mauw, S., Radomirović, S., Trujillo-Rasua, R.: Attack Trees with Sequential Conjunction. In: ICT Systems Security and Privacy Protection. pp. 339–353 (2015)
4. Kriaa, S., Pietre-Cambacedes, L., Bouissou, M., Halgand, Y.: A survey of approaches combining safety and security for industrial control systems. Reliability Engineering & System Safety 139, 156–178 (2015)
5. Leveson, N.: Engineering a Safer World: Systems Thinking Applied to Safety (2011)
6. Leveson, N., Thomas, J.: An STPA Primer (2013)
7. Schmittner, C., Gruber, T., Puschner, P., Schoitsch, E.: Security Application of Failure Mode and Effect Analysis (FMEA). In: Computer Safety, Reliability, and Security. pp. 310–325 (2014)
8. Schmittner, C., Ma, Z., Puschner, P.: Limitation and Improvement of STPA-Sec for Safety and Security Co-analysis. In: Computer Safety, Reliability, and Security, vol. 9923, pp. 195–209 (2016)
9. Steiner, M., Liggesmeyer, P.: Combination of Safety and Security Analysis - Finding Security Problems That Threaten The Safety of a System. In: Workshop on Dependable Embedded and Cyber-Physical Systems DECS of the 32nd International Conference on Computer Safety, Reliability and Security (2013)
10. Young, W., Leveson, N.: Systems Thinking for Safety and Security. In: 29th Annual Computer Security Applications Conference ACSAC. pp. 1–8 (2013)
11. Young, W., Leveson, N.: An Integrated Approach to Safety and Security Based on Systems Theory. Commun. ACM 57(2), 31–35 (2014)