Towards Safety and Security Co-engineering: Challenging Aspects for a Consistent Intertwining

Gabriel Pedroza

Towards safety and security co-engineering
Challenging aspects for a consistent intertwining

Gabriel Pedroza

CEA, LIST
Point Courrier 174,
91191, Gif-sur-Yvette, France
gabriel.pedroza@cea.fr

Abstract. The emergence of systems identified as both safety and security critical has motivated research and industry communities to search for novel approaches to conduct multi-concern engineering (co-engineering). But several aspects and issues have arisen during the process what has limited the advances. Among them, there are the specificities found in concepts, methods and development cycles, the current standalone practices of safety and security, and the lack of consolidated metrics for safety-security assessment. This paper presents synthetic discussions on referred topics along with some suggestions for solutions and perspectives.

Keywords: safety · security · development cycle · co-engineering · MDE · safety-security metrics

1 Introduction

Safety and security are topics often referred in the literature as major concerns to be addressed in systems engineering. Along with the difficulties found in the practice of safety and security in their usual standalone mode, research and industry should also face new challenges arisen from the need of a common practice. The referred need does not only obey to a mere optimization of resources, but it is essentially generated by the emergence - or evolution - of application domains which are identified as both safety and security critical. Indeed, the observed dependencies between safety and security aspects in different use cases, the potential conflicts between proposed solutions, the variety of development and analysis methods, and the growing number of exigencies to improve systems' trustworthiness lead to a singular problematics. Structuring the aspects for a seamless co-engineering process is a vast, complex and, thus, very tough task. This short paper aims to describe, in a non-exhaustive manner, some aspects to move forward, highlight identified issues and perspectives for solutions, and, finally, address some questions that may enrich the ongoing discussions. In particular, it aims to raise attention on the need for a common practice of safety and security via the consistent integration of known techniques.
The rest of the paper is structured as follows. The section 2 presents a conceptual positioning of safety and security. The section 3 gives an overview of the

standards ecosystem. A MDE approach that can be leveraged for safety-security co-engineering is explained in section 4. Some difficulties to achieve integration and adoption of safety and security development cycles comes in section 5. A discussion on the consistent integration of safety and security techniques is given in section 6. Finally, the overall perspectives come in section 7.

2 Positioning safety and security

This section introduces a minimal background to make explicit what safety and security stand for in this paper. To do so, some application domains are recalled including representative use cases which are recognized by the community as both safety and security critical. Afterwards, a conceptual positioning of safety w.r.t. security (or conversely) is given. This positioning helps to highlight commonalities and also specificities of both areas.

2.1 Application domains positioning w.r.t. safety and security

Industry application domains are defined by categories of problematics and by the technology applied/developed to tackle them. A wide variety of systems are developed within each application domain. They include software and hardware artifacts structured by an architecture which is often networked. Despite there is no consensus on metrics to assess criticality, the systems are labeled as critical regarding the goals and missions they should/must accomplish and the potential unwanted impacts of not succeeding in doing so. Several instances of critical systems can be found in aerospace, railway, automotive, health, and nuclear domains but also in other sectors like e-commerce, e-voting, and social-network based systems. The operation of systems in the different domains has been impacted by the evolution of Information and Communication Technology (ICT), and by the progress in autonomy, automation and - more recently - in artificial intelligence techniques. Despite these evolutions, it is likely that the emerging systems will lead to a few new relevant risks, *i*.e., risks that have not been foreseen nor already faced by human-beings. If that occurs, our current paradigms to perceive systems' criticality will remain valid across time irrespective of the type of concern. However, what is changing for sure is the increasing gain in (1) usage of ICT over formerly manual works and missions, (2) systems automation, smartness, and complexity, and (3) physical and virtual accessibility of systems. All in all, the main stake is the trustworthiness that human-beings have on those systems, *i.e.*, the proved reliability, safety, security, etc. of those highly automated-networked-complex systems. By definition, catastrophic risks related to a safety critical system endanger human-being lives [1]. As long as those systems exhibit the three features previously listed, they will also become security critical. Thus for instance, a railway system including ICT artifacts, mostly automated, and physically and virtually accessible will certainly face critical security risks. On the contrary, certain security critical systems will be in no way safety-related, *e.g.*,

e-voting, e-commerce, social network, mobile-communication systems (see Figure 1). This implies that safety and security depend upon technical specificities of application domains. The identification, assessment and management of risks may demand a clear understanding of them.

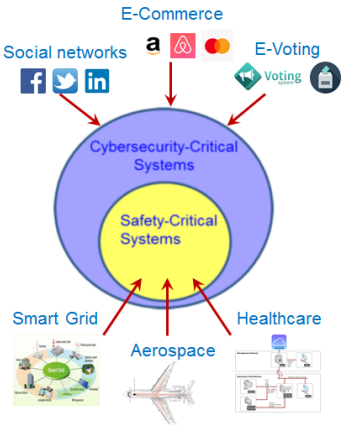

Fig. 1. Relationship between safety and security critical systems

2.2 Conceptual positioning of safety and security

The current standalone practices of safety and security consists in separately integrate them into an engineering development cycle. The outcome is a couple of concern-oriented cycles that allow engineers to separately perform the different development phases up to the system disposal. Figure 2 roughly depicts the current state of safety and security engineering processes and their fundamental elements. Regarding the intersection between safety and security engineering processes, we can highlight the existence of commonalities between both processes. More specifically, there are conceptually similar terms and notions in both areas, like for instance, *risk*, *severity*, *likelihood*, etc. As long as those terms and notions are proven to be semantically related, they constitute a basis for a common safety-security engineering process. But constitute such common basis is rather bulky, in particular, due to the variability of terms and notions found in standards, methods, guides and other technical documents. Some aspects to consider when determining notions similarity are:

Syntactically similar terms with different meanings. It refers to syntactically similar terms used in both safety and security having different meanings. For instance, even if the term *Feared Event* is used in both safety and security risks analyses, it does not necessarily have the same meaning nor structural form. A *Feared Event* in safety can be considered as a combination of a *hazard* and an

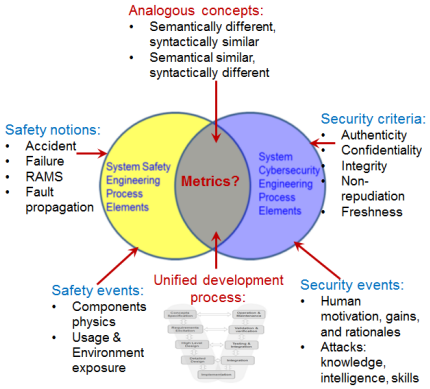

Fig. 2. Conceptual positioning of safety and security elements

operational situation [2] whereas a *Feared Event* in security can be the violation of a *security goal* [3] which is structured by a set of requirements, typed by a criterion or property (confidentiality, integrity, authenticity).

Syntactically different terms with equivalent meanings. In this case, two different terms respectively used in safety and security stand for the same notion. For instance, the term *Threat Condition* introduced in the standards ED202 [5], ED-203 [6] mostly correspond to the notion of *Feared Event* found in standards like ISO-27005 [3] and also in methods like EBIOS [4].

It is expected that a common engineering process shall include metrics useful to evaluate both safety and security risks. Referred metrics are necessary when paths to risks combine both safety and security events, and their likelihood of occurrence and severity of impact need to be evaluated. Along with previous commonalities, several specificities have been already identified:

Specific criteria for evaluation. The technical criteria for evaluation of safety and security are almost specific to each area. Known security criteria are for instance *authenticity, integrity, confidentiality, non-repudiation, freshness, controlled access,* etc. On the other side, relevant safety criteria are for instance *reliability, availability, maintainability,* etc. Even if for some cases certain criteria are common to both areas, for instance *availability*), to our knowledge, most of the criteria remain specific either to security or to safety.

Different nature of events. Safety and security analyses aim to asses the robustness of a system w.r.t. certain unwanted events. However, the nature of those events is rather different for each analysis. As for safety events, we can mention accidents, system failures and functions faults. As for security events, we can mention cyber or physical attacks, *e.*g., intrusions, and intentional damages and failures. Therefore, the physics of components, the system usage and its exposure

to an environment are the root causes of safety events. On the contrary, security events are mostly determined by human-related factors like the motivations, gains and opportunities of attackers. In addition, successful events may require for an attacker to acquire certain knowledge, skills, and resources.

3 Standards, development cycles and methods

3.1 Standards ecosystem

A wide variety of standards have been published targeting safety or security aspects. The Figure 3 shows some of them. Most standards are elaborated not only targeting a given problematics but also considering the specificities of an application domain. In the safety area, the standard IEC-61508 [7] is a generic reference to conduct functional safety analyses of so called Electrical-Electronic-Electronically Programmable (E/E/EP) systems. This generic standard has been taken as a reference in order to adapt and specialize the safety analyses for different application domains. An analogous pattern can be found in the development of standards pertaining to the security area (*e.g.*, considering ISO-27005 [3] as a generic standard). However, the level of maturity, consensus and/or adoption of security standards is still limited and many discussions and work are in progress. This disparity can be partially explained by the late identification of security as a strategic topic in the industry-research landscape. In addition, the analysis of security introduces specificities and new elements which raise questions and increase the complexity of discussions.

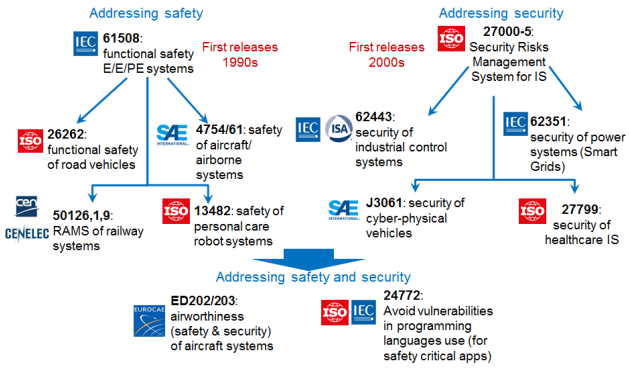

Fig. 3. Overview of some relevant standards for safety and security

Standards not only provide a conceptual basis but also guidelines for the integration of concerns in the development cycle. Thus, development cycles can be accordingly completed via the integration of safety and security aspects. A representative instance of such integrated process can be found in the ISO-26262

standard [2] (safety of road vehicles). A common practice of safety and security engineering will demand and integration of both safety and security aspects.

3.2 Towards an unified development cycle

If a development process integrating safety and security aspects is required, such integration does not suffice to achieve an effective safety-security co-engineering. For instance, typical standalone instances of safety and security development cycles include a requirements elicitation phase. A naive integration can consist in first elicit safety requirements and afterwards the security requirements (or conversely). This sequential integration may result inefficient and may even be ineffective, since it does not address potential conflicts between requirements. In addition, conflicts may not only appear during the problem definition phases but also during the solution phases in the development cycle, *e*.g., a cypher mechanism deployed to protect frames confidentiality may impact system's performance and violate real-time safety constraints. Therefore, a unified development cycle should not only consider the integration of development processes and phases, but also the specific methods and techniques used in safety and security - in addition to the specificities of the application domains. The achievement of such unified development cycle shall be a milestone of industry-research working groups. However its definition is limited by: (1) the observed disparity between security and safety standards (maturity, consensus and adoption), and (2) the complexity of ensuring coherence between safety and security methods and techniques.

4 Model-driven approaches for co-engineering

The importance of integrating not only processes but also methods for safety and security analyses was highlighted in previous section. This section provides insights on model-driven engineering (MDE) techniques for achieving co-engineering of safety and security.

4.1 Safety and security engineering commonalities

The current state of practice shows that safety and security engineering are conducted in standalone mode and guided by independent processes and methods. Figure 4 shows an overview of two instances of processes to conduct safety and security activities in the development cycle. To tackle the structural complexity and integrate both processes into a single one, the following high level commonalities are observed:

- The usage of evaluation criteria is present in both processes and some criteria rely upon discrete evaluation scales (qualitative and quantitative).
- Both processes demand the definition of metrics or scales to evaluate *risks*, *likelihood*, and *severity*.

– Both processes are risk oriented and the evaluation of *risks* is based upon *likelihood* and *severity* scales.
– A function to evaluate the acceptability of *risks* exists in both processes.
– The elicitation of *Feared Events* is targeted in both processes.

Regarding the last item, notice that *Feared Events* identified during the safety analysis might be also targeted by attackers. Conversely, some *Feared Events*, unveiled during the security analysis, might also be caused by purely accidental functional failures. In such cases, common or interdependent *Feared Events* can be the basis to conduct the expected co-engineering.

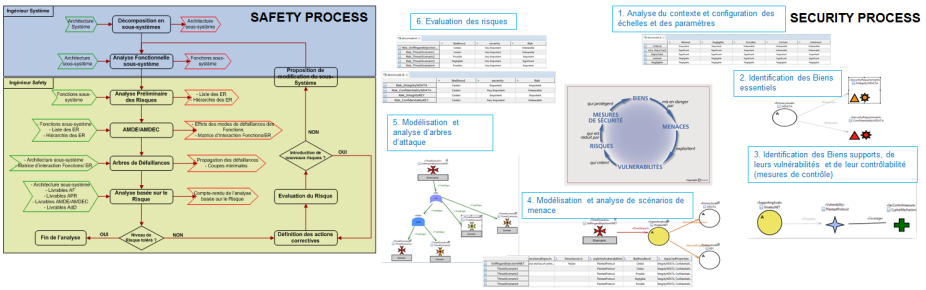

Fig. 4. Overview of standalone safety and security development processes

4.2 Model-driven approaches for multi-concern analyses

It is assumed that the reader is familiar with Model Driven Engineering (MDE) principles and techniques. Non-savvy readers can consult these references [16], [17]. MDE has been applied to support engineers during several phases of the development cycle, *e.g.*, design and verification [8], [9]. In particular, during the design phases a model of the target system is usually constructed. The model is based upon standardized languages like UML [10] and SysML [11] which facilitate typical design tasks like system structuring, refinement, decomposition, extension, and/or transformation. MDE languages are flexible enough to be extended and specialized so as to capture the elements necessary to conduct safety and security analyses. Indeed, fundamental concepts, the relationships between them, and analyses steps can also be represented and implemented. Following typical MDE approaches, the system model can be enriched either with the elements related to safety or security. Annotating finally yields two models suitable to -separately- conduct safety or security analyses (see Figure 5); the annotated models are insufficient to support joint safety-security analyses. However, they provide a basis to construct the co-engineering framework as explained in next subsection.

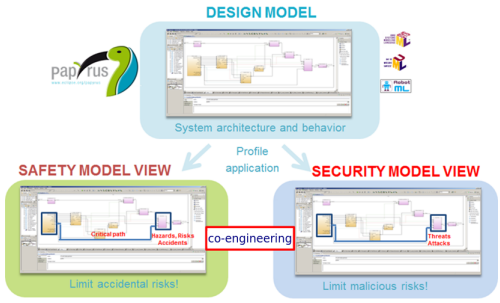

Fig. 5. View of safety and security models obtained after annotations

4.3 Joint safety-security engineering

To support joint safety-security analyses (co-engineering), the MDE framework can be leveraged in the following way:

Integration of meta-models. Safety and security annotations are defined in separated meta-models. The meta-models allow to capture the fundamental notions, principles associated to the concern and also capture their relationships. The conceptual alignment, necessary to achieve joint safety-security analyses, can be initiated by identifying common elements, prior to a first integration of meta-models. The Figure 6 shows an excerpt of a diagram used to associate concepts pertaining to different standards.

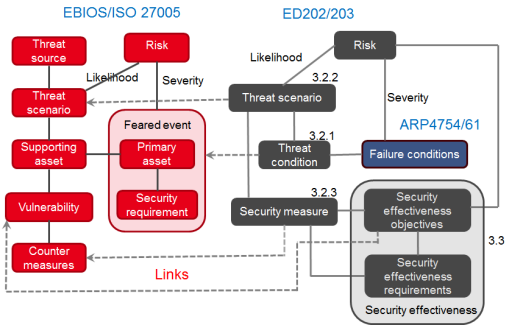

Fig. 6. Excerpt of a diagram showing meta-models integration

Solving conflicts. Once meta-models are aligned by linking common parts, the dependencies can be observed and the potential conflicts along methods phases can be better identified and solved. Recall that conflicts can appear from the application of safety and security methods and techniques and also between their outcomes. In particular, when a technique is applied over modeling instances that are out of common parts in the integrated metamodel.

Integration of processes and phases. Once method phases are supposed to be "free of conflicts", the processes can be integrated. Notice that certain conflicts can not be identified in advance at method level. For instance, conflicts between safety and security requirements - elicited at different phases - can appear after a first architecture design is made. A safety-security development cycle should consider this kind of issues.

Framework implementation. The framework supporting joint safety-security analyses can be implemented following typical MDE development steps: metamodel implementation via a UML/SysML profile, generation of profile code, implementation of safety and security algorithms, customization of the framework front-end, and building the tool product.

5 Difficulties to achieve integration of safety and security processes and their adoption

In previous section, a generic, coarse description was provided about leveraging MDE techniques in order to achieve safety-security co-engineering. Our involvement in several academy-industry projects (AMASS, SESNA, ModSécAéro) allows us to ensure the effectiveness of those MDE techniques. By doing so, several technical and non technical difficulties have been identified which may limit the progress of ongoing work. First, regarding the current state of standalone safety and security development processes, they are in general complex, costly and mostly human based: few support for tasks automation exists. Despite there exist development processes that show certain integration of safety and security aspects (*e.g.*, [5]), they are rather specific to an application domain. In general, a low level of maturity is observed in aspects like integration, tool support and automation. Other aspects impacting the dissemination and progress of safety-security co-engineering are related to current engineering practices. Globally, conduct joint safety-security analyses is a quite recent research and engineering area. To our knowledge, information about case studies showing co-engineering in practice are rare. In addition, there is almost no published feedback from industry on the effectiveness of applied methods and techniques. It is reasonable to believe that some years are still necessary to consolidate our understanding on the topic. For many industry sectors and several application domains, cyber-security is almost an emerging concern. In those cases, an *a posteriori* reaction to cyber-security threats is observed: the lack of awarness on cyber-security risks/culture often leads to underestimate, misunderstand or believe not concerned by the potential threats. Some market, organizational and engineers practices may facilitate the adoption of safety-security engineering processes. In particular, certification is a powerful mechanism to align technical criteria and ensure trustworthiness what finally shapes markets. However today, no certification process for safety-security critical systems exists so far. Last but not least, the impact of safety and security concerns in current organizations and engineer practices should be better identified in order to facilitate the adoption of an integrated development cycle.

6 Integration of safety and security techniques

This section is dedicated to explain an instance of techniques integration. The chosen techniques are usually applied in safety and security. The instance is relevant since it helps to highlight some challenges for a consistent safety-security intertwining.

6.1 Combined attack-fault trees

Fault and attack trees are known techniques respectively used in safety and security areas. They are means to structure and evaluate unwanted events impacting a target system. On one side, fault trees are often composed by nodes representing system failures as boolean variables. The fault nodes are linked by logical gates AND, OR and can be assigned with a probability of occurrence. On the other side attack trees can be composed by nodes representing vulnerable states of the system, attacker actions, or conditions for attack progression. They are also linked by logical gates AND, OR. Despite the assignation of probabilities to attack nodes has been suggested, the estimation and interpretation of outcomes are still arguable (more details in subsection 6.4). Several approaches have been proposed to integrate (merge) fault and attack trees, *e.*g., [13]. However, the integration is mostly structural and, in general, several issues still remain unsolved. Some pros and cons observed in approaches for tree merging are described in the following items:

Pros: Most algorithms for integration of fault and attack trees have a polynomial complexity on the number of nodes. Those algorithms can be implemented thus providing support for automatically merging trees. In certain cases, the attack tree is transformed towards a fault tree yielding a merged tree with a simpler structure. It is also observed that safety metrics and functions are often reused for evaluating certain properties of attacks. In particular, the failure rate and the Mean Time to Failure inspire their security counterparts, namely, Attack rate, and Mean Time to Attack. These metrics and functions are useful to estimate the probability of attacks occurrence.

Cons: The variety of attack tree nodes semantics seems broader than their safety counterparts. Indeed, irrespective of the reference considered, the definition of fault trees remains mostly equivalent. On the contrary, there is no common definition for attack trees and consequently a rather heterogeneous variety of definitions exist. No common semantic for attack tree nodes can be identified so far. As a consequence, nodes describing attack steps or actions, can be specified at different levels of abstraction and granularity. Since vulnerabilities can be present at different system levels (HW/SW) and caused by different types of flaws, the tree nodes representing them are also heterogeneous. Attack nodes representing the conditions for attack progression show similar characteristics. Referred specificities suggest that, to keep consistency, fault and attack trees merging shall mostly remain a human-based

task. Since the semantics and nature of nodes (safety and security events) are different, reusing safety metrics and functions to evaluate security aspects should be more thoroughly considered.

6.2 Discussion on metrics for safety assessment

The so called Mean Time Before Failure (MTBF) is a central metrics applied in safety to estimate systems life and other related features. The MTBF is computed by obtaining the mathematical mean of a probability distribution function (*pdf*) with exponential basis (see Figure 7). The exponential *pdf* measures the probability of failures occurrence and is characterized by its parameter λ: λ is a failure rate which measures the failures of a component (or system) per time unit. However, as shown in Figure 7, the exponential-based *pdf* is not the only probability distribution available.

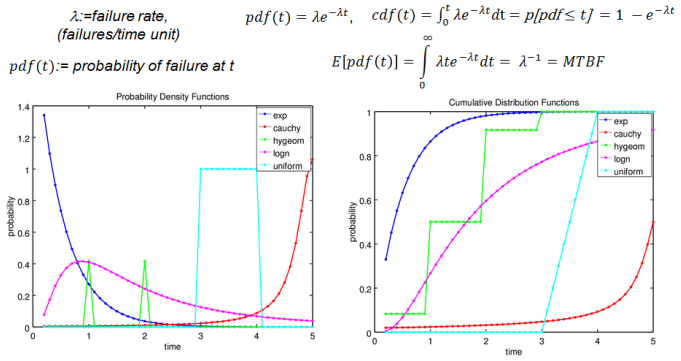

Fig. 7. Some metrics and functions used for systems safety assessment

The exponential *pdf* is currently adopted as a valid metric thanks to (1) the experiments that can be conducted to compute λ and (2) the effectiveness of MTBF predictions w.r.t. systems life observed at field. These facts imply that a suitable correspondence between the mathematical model and the physical phenomenon (systems ageing) has been settled. One of the main hypothesis for the estimation of the MTBF is that the failure rate λ remains constant. However, the validity of this hypothesis is limited. The Figure 8 depicts the actual evolution of components failure rate across their life time. It is observed that the failure rate is mostly constant during a life interval (called normal), but rapidly changes during youth and ageing stages. All in all, even if the mathematical model reflects the essence of a phenomenon, it remains constrained by the validity of hypotheses. Recent experimental results show that certain MTBF predictions may differ from real life time of systems observed at field [14]. According to this study, the failure rate λ is not only determined by the physics of components, their nominal usage, and the exposure to a given environment. It also depends upon

the quality of the process for components (and subcomponents) development. Moreover, accidental damages, occurred during manufacturing, may over-stress the components and finally increase their failure rate. A more precise calculation of λ shall require to consider previous factors.

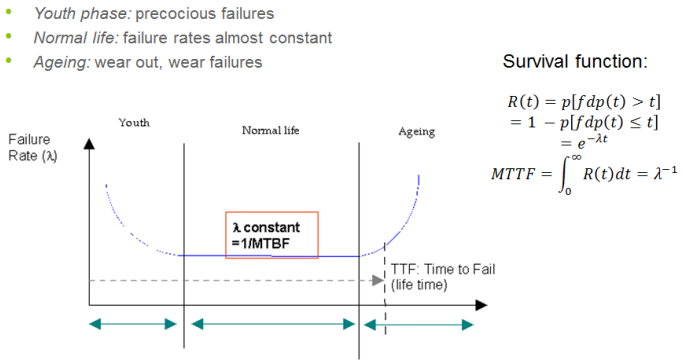

Fig. 8. Typical evolution of failure rate across time. The graph is borrowed from [14]

6.3 Discussion on metrics for security assessment

Many approaches found in the literature rely upon variants of the exponential-based *pdf* to estimate the probability of an attack. However, not many address the question about the adequacy of this mathematical model w.r.t. the phenomenon, *i*.e., the attack progression. The work in [12] provides experimental results on the time to compromise a large informatics system. The proposed metrics is named Mean Time to First Attack (MTFA) and is calculated from data gathered from intrusions at field. The collected data are used to compute attacks frequency and afterwards to calculate parameters of several *pdf*'s (and in particular λ). The predictions of the MTFA relying upon different *pdf*'s are then compared to the real periods of attacks' occurrence. The results of the comparison show that the best fitted model is not the exponential but the Pareto based *pdf* [12]. Thus, it is reasonable to question about other potential mismatches like the one just identified for the safety assessment metrics. In particular, whether the attack rate remains constant along a given period during the systems life. Or even whether the use of the attack rate suffices to characterize the whole phenomenon which involves threats and vulnerabilities. A first element to answer these questions is that, along with factors affecting elements manufacturing (quality of development process, accidental damages), the attack rate is likely also impacted by other factors appearing before and during attack execution. Those factors could be identified and analyzed by considering the phases of attack preparation and deployment. The so called *intrusion kill chain* [15] (see Figure 9) defines several attack phases which can be useful for that purpose.

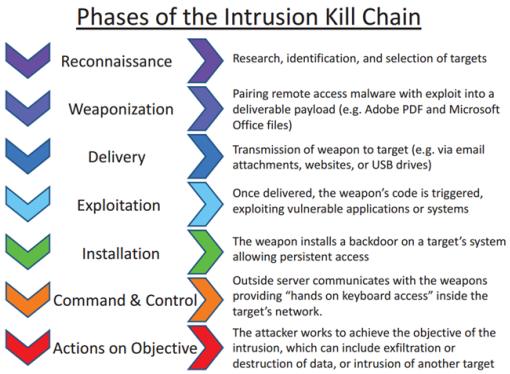

Fig. 9. Phases of the so named *intrusion kill chain*. Image borrowed from [15]

6.4 Perspectives for consistent assessment of safety and security

The use of field data to validate a mathematical model for attack prediction seems a consistent approach. However, the metrics used for security assessment may need to be validated in larger case studies and for other kind of systems. The *pdf*'s for predicting security (and also safety) events occurrence might be better rely upon other basis than the exponential. Further studies may help to identify the accuracy of predictions already obtained with the exponential model. To gain representativeness in the security assessment, larger and more diversified field data are necessary, for instance, data from different attack categories, known vulnerabilities, and application domains. To improve the computation of attack rates, factors related to attack preparation and deployment phases need to be introduced, for instance, attacker resources, skills, smartness, and motivations. Nonetheless, increasing the accuracy of a mathematical model also increases its complexity, and the cost and complexity of the prediction method. Consequently, more guidance and support will be necessary to bring forward these suggestions.

7 Overall perspectives

Safety and security co-engineering is rather a young area and further works and progress are expected to gain in maturity. Regarding the conceptual and processes integration, certain engineering techniques (like MDE) will contribute to consolidate the the co-engineering of safety and security. However, certain specificities of each area will remain and it is likely that standalone development practices will prevail for a while. The choice of a unified safety-security development cycle seems feasible but its deployment and adoption remains, for now, complex and costly to achieve. Further methodological and tool support are needed to help concerned communities to overcome these issues. Regarding the metrics for safety-security assessment, the consistency and validity of metrics need to be ensured in order to achieve more accurate predictions. The application of mathematical models in security assessment still needs further validations.

References

1. SAE International : ARP4754A - Guidelines for Development of Civil Aircraft and Systems. SAE International, https://www.sae.org/standards/content/arp4754a/, 2010.
2. International Organization for Standardization: ISO 26262 - Road vehicles - Functional safety. ISO, https://www.iso.org/standard/43464.html, 2011.
3. International Organization for Standardization: ISO 27005 - Information technology - Security techniques - Information security risk management. ISO, https://www.iso.org/standard/75281.html, 2018.
4. Agence Nationale de la Sécurité des Systèmes d'Information: EBIOS - Expression des Besoins et Identification des Objectifs de Sécurité. ANSSI, https://www.ssi.gouv.fr/guide/ebios-2010-expression-des-besoins-et-identification-des-objectifs-de-securite/, 2010.
5. European Organization for Civil Aviation Equipment: ED202 - Airworthiness Security Process Specification. EUROCAE, https://eurocae.net/, 2014.
6. European Organization for Civil Aviation Equipment: ED203 - Airworthiness Security Methods and Considerations. EUROCAE, https://eurocae.net/, 2018.
7. International Electrotechnical Commission: IEC 61508 - Functional safety of electrical/electronic/programmable electronic safety-related systems. IEC, https://webstore.iec.ch/publication/22273 2010.
8. G. Pedroza, M. S. Idrees, L. Apvrille and Y. Roudier: A Formal Methodology Applied to Secure Over-the-Air Automotive Applications. In: Vehicular Technology Conference (VTC Fall) on Proceedings, pp. 1–5. IEEE, San Francisco, CA, 2011.
9. B. Hamid, S. Gürgens, A. Fuchs: Security patterns modeling and formalization for pattern-based development of secure software systems. Journal Innovations in Systems and Software Engineering, vol. 12, no. 2, pp. 109–140. Springer, London, 2016.
10. Object Management Group: Unified Modeling Language specification. OMG, https://www.omg.org/spec/UML/About-UML/, 2017.
11. Object Management Group: System Modeling Language specification. OMG, https://www.omg.org/spec/SysML/About-SysML/, 2017.
12. H. Holm: A Large-Scale Study of the Time Required to Compromise a Computer System. In: Transactions on Dependable and Secure Computing on Proceedings, IEEE, vol. 11, no. 1, pp. 2–15, Jan.-Feb. 2014.
13. Igor Nai Fovino, Marcelo Masera, Alessio De Cian: Integrating cyber attacks within fault trees. In: Reliability Engineering & System Safety, ScienceDirect, LNCS Elsevier, vol. 94, issue 9, pp. 1394-1402, 2009.
14. Institut pour la Maîtrise des Risques: Experimentation of the new Reliability prediction method FIDES. IMDR, https://eeepitnl.tksc.jaxa.jp/mews/en/20th/data/1_10.pdf, 2017.
15. U.S. Senate-Committee on Commerce, Science, and Transportation: A "Kill Chain" Analysis of the 2013 Target Data Breach-March 26 2013, https://www.commerce.senate.gov/public/, USA, 2014.
16. Object Management Group, https://www.omg.org, 2019.
17. The Eclipse Foundation: Papyrus, https://www.eclipse.org/papyrus/, 2019.