Similarities and dissimilarities between safety levels and security levels

Jean-Paul Blanquart, Pierre Bieber, Gilles Descargues, Eric Hazane, Mathias Julien, Laurent Léonardon

Similarities and dissimilarities between safety levels and security levels

Jean-Paul Blanquart[(1)], Pierre Bieber[(2)], Gilles Descargues[(3)],
Eric Hazane[(4)], Mathias Julien[(5)], Laurent Léonardon[(6)]

(1) Contact author, Astrium Satellites, jean-paul.blanquart@astrium.eads.net
(2): Onera; (3) Thales; (4) EADS Cassidian Apsys;
(5) Altran (on behalf of Airbus); (6) Rockwell Collins France

Keywords: Safety, security, assurance levels, standards

Abstract

The paper proposes a comparative analysis of the notions of Safety Levels and Security Levels as defined (under various names) by the relevant standards. This comparison is a basis for the elaboration of a harmonised process to develop and validate embedded systems having to comply with both safety and security requirements (including related certification requirements when applicable), which is the objective of the French collaborative project SEISES. An important case corresponds to systems for which security requirements come from safety needs i.e., the necessity to preserve safety properties even in case of security threats. In such a case it is necessary to identify clearly the dependencies between the Safety and the Security Levels of the system.

1. Introduction

Qualitatively ordered scales of levels (designated using various names such as categories, classes, etc.) have been introduced and are usual practice, recommended or required by standards in many domains and especially for safety and for security. The objectives of these scales are to support the identification and adaptation of at least a part of the appropriate development and assurance means and requirements with respect to at least a part of the needs.

For safety concerns, in aeronautics and space as well as many other sectors, some guidelines, tools and common methodologies are accepted both by manufacturers (and suppliers) for products development and (Airworthiness) authorities for certification process.

For security concerns, the situation is quite different: processes, tools and methodologies do exist but often in the typical frame of companies Information System Security, not to operate securely for instance a commercial aircraft carrying several hundreds of passengers! In other words, the situation is less simple. One important reason is that in this case the failure occurrence process (on which one may expect to act through the appropriate development and validation means) cannot be fully derived in a quite straightforward way from an analysis of the needs, failure impact etc. It also depends on some intrinsic characteristics of the system or item, and characteristics of its environment, that are not fully controllable or predictable.

Actually, the security properties of a system strongly depend on a smart combination between risks acceptance (based on a risk analysis process such as ISO 27005 [1]), cost-effective measures implementation (technical, organizational and processes) and an evolving threat environment / security context (attackers technical potential, motivation, skills, prepare-readiness, etc.).

The objectives of the SEISES project are to elaborate a harmonised process to support the development and validation of systems with both safety and security requirements in a cost effective manner, and in particular to support the elaboration of the proper justification evidence as required by the applicable safety and security standards and authorities [2]. An important foundation and prerequisite of this harmonised process is the definition of Safety/Security Level(s), based on analysis and studies, which could be accepted by the whole Safety/Security community and Airworthiness authorities.

2. Safety Levels

For what concerns safety, in aeronautics and space as well as many other domains, there is a complete safety-level[1]-based framework with a series of scales from the most abstract safety needs down to the most detailed development and assurance requirements and means, and a process to allocate the various entities addressed along the design to the so defined levels or categories.

For civil aviation the ARP 4754 / ED 79 [3] classifies "failure conditions" i.e., hazardous situations resulting from potential failures of the aircraft functions. Then Development Assurance Levels are assigned depending on the severity classification of Failure Conditions considering the possible independence between development processes that can limit the consequences of development errors. The more severe the Failure Condition Classification, the greater the level of Development Assurance necessary to mitigate the Failure Condition. The classification of each failure condition is based on its identified effects. Five categories are defined and ordered, labelled Catastrophic, Hazardous, Major, Minor and "no safety effect", corresponding to predefined descriptions of possible effects (multiple fatalities, reduction of the capabilities to cope with adverse situations, etc.). For each category there is a one-to-one mapping to a scale of "Development Assurance Levels" (DAL) labelled from A (most demanding, corresponding to catastrophic failure conditions) to E (least demanding, corresponding to "no safety effect").

The Development Assurance Level assignment (allocation) process begins with assignment to the Functions involved in the Failure Conditions based on its most severe Top-Level Failure Condition Classification and in accordance with Table 1 below.

Failure Condition Class	Quantitative Safety Requirement (failures/h)	Development Assurance level
Catastrophic	$P < 10^{-9}$	A
Hazardous	$P < 10^{-7}$	B
Major	$P < 10^{-5}$	C
Minor	None	D
No safety effect	None	E

Table 1: Severity and Development Assurance Levels (ARP 4754 / ED 79 [3])

At this stage it is possible to refine the assignment to functions taking into consideration the dependability architecture based on various redundancy schemes and proper justification of independence with respect to potential faults[2].

Development Assurance Levels are finally assigned to the items implementing the various functions, based on the most demanding classification of the implemented functions.

For space systems, in Europe the ECSS (European Cooperation for Space Standardisation) standards for safety [4] and dependability [5] define a very similar approach where the source of categories as the end effects of the potential failures of the considered space system. These failures are categorised according to a ranking of the severity of their consequences, according to a four-level scale. The allocation process starts at system level by the classification into categories of the system functions, based on the highest severity of the consequences that could result from their failures.

---

[1] We use here the expression "safety level" to designate the general notion identified in various domains by Safety Integrity Level, Automotive Safety Integrity Level, Criticality Category, Development Assurance Level etc.

[2] This part of the process is sometimes called "DAL downgrading" which may be misleading. Indeed the objectives are not to assign a lower DAL than needed, but to acknowledge the fact that when properly justified, the really needed DAL may be lower than what results from the basic primary assignment rule.

These categories, for functions, are called “criticality categories”. It is worth noting that the same allocation process is applied also to the operations.

Once the system functions (and operations) are categorised, this process being iterated along functional decomposition, the hardware and software products are themselves classified into categories, based on the simple general rule stating that a product is allocated the category corresponding to the highest criticality function among the possibly several functions associated to that product.

It is worth noting that the ECSS standards do not describe explicitly how and to what extent it is possible to take into consideration the dependability architecture and adapt the allocation of safety categories to the elements of the system. This process is only supported by the general allocation rule. It is therefore necessary to analyse and justify on a case by case basis the propagation and impact of potential failures, taking into account the dependability architecture and mechanisms and the necessary independence arguments.

In other domains (e.g., automotive, industrial automation, nuclear, railway) a similar approach is defined even though there exists some differences as shown in comparative analyses such as [6]. All domains share the same fundamental basis where the categories, the safety levels, represent the risks associated to the end effects of the potential failures of the considered system. Risks are classically measured by a combination of their severity and occurrence probability or likelihood. The fact that some domains focus more on the occurrence (e.g., railway, industrial automation) or on severity (e.g., space), or more explicitly on their combination, may be seen as a presentation choice. Indeed all domains rely on a similar scheme where there is one-to-one mapping between the severity and the maximum probability corresponding to the risk acceptability frontier for that severity. Therefore these notions can be considered as equivalent, provided the acceptability frontier is well defined.

In most domains, the first categorised element of a system is a (top level) function, which inherits its category from the category of the risk induced by its potential failures. Then the categories are derived following the functional decomposition and finally allocated to the elements implementing the functions, taking into account dependability architecture considerations when properly justified.

Finally, even though the Safety Level definition and allocation process is based on some assumptions always worthwhile to discuss and challenge, and present some differences among domains, there is a general consensus on the general and simple approach, based on the idea that at least for what concerns accidental faults (be they random hardware faults or “systematic” design faults e.g., in system or software) it is possible and meaningful to define a mapping between:

- On the one hand, the “level of trust” one should have on the fact that the considered system (or entity, etc.) will not fail (and this level can be directly mapped to a level of safety need, taking into consideration the severity of the end effects of potential failures of the considered system or entity, and some agreed rules on risk acceptability establishing constraints between the occurrence and the severity parts of the risk),
- On the other hand, the development and assurance means and requirements, covering:
    - Random hardware faults through component selection, architecture and fault tolerance and probabilistic assessment through agreed models and data,
    - Design faults through levels of rigour in the requirements made applicable to the development and assurance (for instance for the software as described in corresponding standards for aeronautics [7] or space [8]).

An important consequence is that a large part of the safety requirements (the so-called “non-functional” safety requirements) may be managed in a generic way through the notion of Safety Level which gathers all that one needs to know so as to identify from an analysis of the top-level safety needs (added to architectural considerations which impact the allocation of the Safety Level), which requirements are applicable.

3. Security Levels

For what concerns security (and also the part of the safety related to intentional faults) the situation is less simple. Indeed The notion of Security Level encompasses many aspects that can be illustrated in the Figure 1 below.

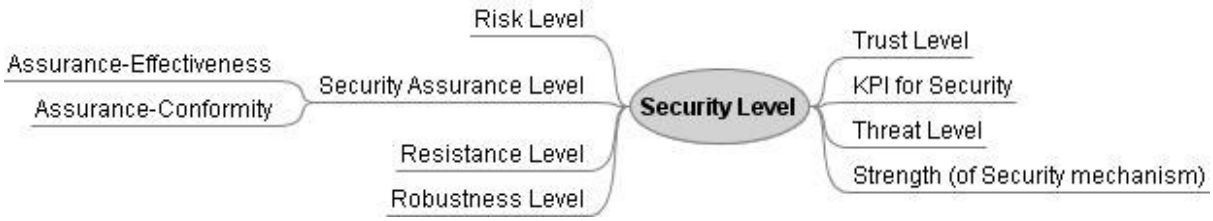

Figure 1: Elements of a Security Level

It may be noted in particular that if the Security Level is, in the same way as for safety, related to notions such as e.g., Risk Level and Trust Level, there are also some other elements that may not have all an immediate counterpart for safety (e.g., threat level, resistance level etc.).

In the paper, we analyse and discuss the various notions related to the Security Level. We focus in particular on an important point, which leads to consider two components of the Security Level, related to Effectiveness and to Conformity.

Indeed, in the case of security, the failure occurrence process (to which constraints ("level of trust") can be derived in a quite straightforward way from an analysis of the needs, failure impact etc.) depends, in addition to some intrinsic characteristics of the system or entity (on which one can act through the appropriate level of development and assurance), also and importantly to some other characteristics out of which not all are fully controllable or predictable. In particular the "ultimate" security properties of a system depend strongly on the characteristics of potential attacks (likelihood and "strength", themselves depending on attacker motivations, skills, means, disclosure of information about the system or its vulnerabilities, etc.).

3.1 Security Level Allocation Process

As in the safety process, a Security Level can be allocated to functions at the end of the risk analysis process. The security risk analysis process is made of the following main steps: context establishment, risk identification, risk estimation, risk assessment, risk treatment and risk acceptance.

Context establishment first defines the perimeter to be taken into account during the analysis. Then, risk identification lists the assets e.g. the resources that need to be included in the security risk assessment. In industry standards for information security, an asset is a resource of value to the organization. In airworthiness security, the assets are those portions of the equipment which may be attacked with adverse effect on airworthiness. Threat Conditions are defined. They are events that have an adverse effect on the security of the assets. Then, the design is analyzed in order to find threat scenarios leading to the Threat Conditions.

The risk of a Threat Condition is unacceptable when an organization or agency will not tolerate its likelihood, e.g. the number of times that Threat Condition can be expected to occur over the span of operation without further mitigation. The risk of a Threat Condition is computed by aggregating the risks of the Threat scenarios leading to this Threat Condition. If the risk is found to be unacceptable, risk mitigations should be defined to modify or augment the design. The risk acceptability should then be reviewed after updating the risk assessment to consider the modified design. This should be repeated until an acceptable risk is found.

The security risk estimation and assessment characterize the threat scenarios with the assets impacted by the scenario, the threat conditions that are reached by the scenario, the vulnerabilities used by the attacker to perform the scenario and existing security countermeasures.

Risk treatment may change the design, or add security countermeasures which have sufficient effectiveness to mitigate the risk to an acceptable level. The Security Level is used to classify the effectiveness required to reach an acceptable level. Any coherent set of features or functions which is used to mitigate attacks may be a security countermeasure, even if it was not intended primarily for that purpose. A security countermeasure can include aircraft equipment or organizational procedures of the external dependencies (features, policies, or procedures documented in user guidance or external agreements).

As defined in e.g., [9], the Security Level associated with a function can be seen as representative of its ability to reduce the likelihood of success of attacks, which is generally decomposed in two parts:

- Effectiveness, or strength, characterising the ability of a security mechanism to protect against an attack, when working properly,
- Conformity, characterising the trust in the ability of a security mechanism to work as intended.

A Security Level ranging from E to A can be allocated to functions that occur in a threat scenario. The Security Level allocation rule is based on the reduction of the likelihood of the threat scenario to an acceptable level. In ED202, five likelihood values are considered: frequent, probable, remote, extremely remote and extremely improbable. For instance, let's consider an architecture containing a function that counters a threat whose likelihood is frequent. If this function is able to reduce the likelihood of the threat from frequent to extremely remote then it would be associated SL B.

Likelihood Reduction	0	1	2	3	4
Security Level	E	D	C	B	A

Table 2: Security Level allocation rule

The security level is used to select adequate assurance requirements both in terms of effectiveness and conformity. Interesting assurance requirements can be found in the Common Criteria.

3.2 Common Criteria Evaluation Assurance Levels

The Common Criteria [10] methodology is designed to specifically address security functionality and assurance of that functionality. The CC organizes assurance requirements into classes which identifies the general topic covered by that class. The 6 classes of Security Assurance Requirements (SAR) are:

- ALC: Life-Cycle Support: activities related with development support (including la configuration management, implementation tools and standards, development environment, product delivery)
- AGD: Guidance Documents: activities related with the preparation of guides and manuals for the secure installation and use of the product
- ADV: Development: activities related with the development steps. The development of the product follows the main steps described in the following figure. As in the case of ED202, activities deal with the development of security functions rather than the development of all functions of the product (as in the case of DO178B).
- ASE: Security Target Evaluation: activities related with the content of the security target (security target description, security objectives, security functions and assurance activities, …)
- ATE: Tests: activities related with product testing (functional test, independents tests performed by the evaluator, coverage and depth levels)
- AVA: Vulnerability Assessment

Each class is decomposed in several families. For instance ADV class has 6 families (ADV_ARC, ADV_FSP, ADV_IMP, ADV_INT, ADV_SPM, ADV_TDS). The ADV_ARC and AVA_VAN families contain assurance requirements related with effectiveness whereas other families are related with conformity. Each family has one or more assurance component. For instance, ADV_ARC has one component whereas ADV_FSP has 6 components. Assurance components are leveled depending on either the level of detail and rigor used to describe the elements that are required to perform the evaluation, or the scope of the elements described for the evaluation. An Evaluation Assurance Level (EAL) defines the package of components that have to be applied in each family. The CC thus offers different levels of assurance: the EALs range from EAL1, the lowest level of assurance to EAL7, the highest security assurance level.

3.3 Mapping SL and EAL

Unfortunately, the CC does not provide guidance on how to allocate the EAL. We investigated the relationship between Security Level and EAL. A basic way to map a SL and an EAL is to use a table such as table 2 that tells if a function is allocated SL C then all activities related with EAL A should be performed. One feed-back from our ITSEF partner is that High-level CC evaluations (evaluations above EAL5) are not frequent at all. Hence it would not be realistic to include levels EAL6 and EAL7 in the mapping table.

We proposed to directly link the SL with an AVA_VAN component. This is justified because the primary focus of the SL is the effectiveness of the security function and AVA_VAN represents the effectiveness assurance activities of the CC. A feed-back from our ITSEF partner on this proposal was that AVA_VAN.5 considers an attack potential that would require heavy equipment and a time-consuming analysis. Such analyses are relevant in the domain of hardware products where the operational environment is very hostile (e.g. smartcards). Most of the targeted products addressed by the SEISES project include embedded software running on a complex or proprietary platform. So it was proposed to limit the AVA_VAN to AVA_VAN.4. This is consistent with the proposal to exclude EAL6 and EAL7 from the mapping.

The allocation of an AVA_VAN level also defines a number of applicable assurance-correctness activities because they need to be performed prior to performing AVA_VAN. For instance, to perform AVA_VAN.3 you need to have previously performed ADV_ARC.1, ADV_FSP.4, ADV_IMP.1, ADV_TDS.3, AGD_OPE.1 and AGD_PRE.1. Furthermore, an AVA_VAN component is related with one or several EAL. For instance, AVA_VAN.3 is related with EAL4. This relation defines another set of applicable assurance-correctness activities. For EAL4, components in classes ALC and ATE should also be performed.

The table 3 below shows a possible mapping between SL and EAL where the AVA_VAN component and the SL are increased consistently. One drawback of this table is that there is no distinction between SL A and SL B.

SL	E	D	C	B	A
AVA	1	2	3	4	4
EAL	1	2,3	4	5	5

Table 3: A possible SL/EAL Mapping.

It is worth noting that this scheme leaves open the issue regarding the mapping between Security Levels "on the side of the means" and "on the side of the needs". Both components of the Security Levels can be mapped to the development or assurance means and requirements, but the end effect on the service properties still depend on the likelihood of attacks and their nature ("attack potential").

4. What could be a "Security-for-Safety Level"?

There is a large class of systems for which safety and security needs are not independent and in particular the safety systems for which the safety properties must be ensured even in the presence of security threats. For these systems, at least a part of the security needs results in fact from the safety needs (in what is generally called "security for safety").

ED202 focus on "security for safety". Consequently, it relates the security and safety processes at several stages.

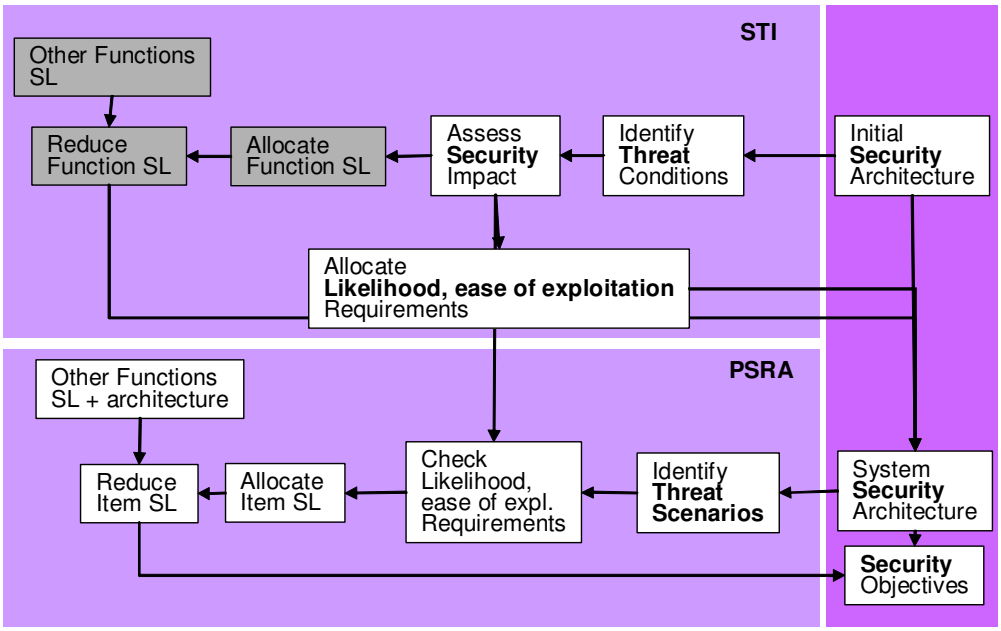

Figure 2: SL Allocation process

During risk identification, only Threat Conditions with a safety impact are considered. So Threat Conditions should be related with at least one of the Failure Conditions defined by the Safety Hazard Assessment. At this step one difference between both processes is: a SL is not allocated to functions on the basis of the analysis of Threat Conditions whereas a DAL may be allocated to functions on the basis of the severity of the Failure Conditions related with this function failure. In ED202, the allocation of SL is based on the reduction of likelihood of Threat Scenarios that can only be defined when the architecture is known. So the activities “allocate SL” and “reduce SL” are colored in grey in the previous picture that summarizes the main steps of the SL allocation process.

Proba (per Flight Hour)	[1, 10^{-3}[	[10^{-3}, 10^{-5}[	[10^{-5}, 10^{-7}[	[10^{-7}, 10^{-9}[	[10^{-9}, 0]
Likelihood	frequent	probable	remote	ext. remote	ext. improbable

Table 4: Likelihood and Quantitative Safety Requirements

In ED202 the likelihood is defined as “*the number of occurrence of threat scenario related to the life of an aircraft*”, this is identical to the definition of average probability that is used in safety. Furthermore, the likelihood values used in ED202 are similar to probability intervals used in the safety process to associate quantitative requirements for accidental faults.

Another important connection between safety and security is the security risk acceptance matrix that mimics the safety risk acceptance matrix. In the following table green cells (risk is low) describe acceptable pairs of Threat Condition Likelihood and Impact Severity.

Risk Level		Impact				
Threat Likelihood		V	IV	III	II	I
		No Effect	Minor	Major	Hazardous	Catastrophic
pV	Frequent	Low	Medium	Medium	High	High
pIV	Probable	Low	Low	Medium	Medium	High
pIII	Remote	Low	Low	Low	Medium	Medium*
pII	Extremely Remote	Low	Low	Low	Low	Medium*
pI	Extremely Improbable	Low	Low	Low	Low	Medium*

*** = Risk justification must include demonstrating the absence of a single point of security weakness**

Table 5: Risk Matrix (from ED-202 v2)

Analysis performed in order to allocate DAL and SL are quite different and are not directly connected. DAL allocation is based on a purely qualitative analysis of the minimal cut sets whereas SL is based on a quantitative analysis (likelihood reduction) of the threat scenarios.

Another difference between DAL and SL is that the DAL level of a function is directly determined by the safety impact of a development fault for this function. Before DAL refinement rules (see section 2 and note 2) are applied the following property is true: the most severe the safety impact is, the higher the DAL level will be. In the case of SL, it is not clear whether a similar property holds.

Consider a function involved in a threat scenario leading to a Threat Condition related with a Catastrophic Failure Condition. If this function reduces the likelihood of the threat scenario by one level (from extremely remote to extremely improbable) it will be allocated SL D, this is a rather low level. Suppose now that the Failure Condition is less severe, it is Hazardous and the function reduces the likelihood of the threat scenario by three levels (from frequent to extremely remote) it will be allocated SL B, this is a rather high level. This example shows that the safety impact of a security function is more directly related with its DAL than with its Security Level.

One explanation of this difference is that the SL aims at defining a requirement on the effectiveness of the security function whereas the DAL focus is on the implementation assurance aspects. In the previous example, the security function would be allocated DAL A in the first case and DAL B in the second one. So the development of the security function should be able to cope with pairs of levels such as (DAL A, SL D) meaning that the development shall reach a high level of implementation assurance but a rather low level of effectiveness or (DAL B, SL B) meaning that the development shall reach a high level on both aspects.

When DAL refinement rules are applied then the difference is less obvious as the dependence between a function DAL and its safety impact is less direct. In that case the development could have to deal with pairs of levels where the DAL is smaller than the SL. This kind of pairs does not seem to be consistent because it combines a requirement for high effectiveness and low implementation assurance and it seems obvious that an incorrect implementation of the function would destroy its effectiveness. To avoid this situation we could draw a parallel between effectiveness and reliability and adapt an ARP4754 allocation rule that requires that, when several functions contribute in conjunction to a Failure Condition, the most reliable function is allocated the highest DAL. We could require that the most efficient function (e.g. the function with the highest SL) should be allocated the highest DAL.

The SEISES project targets the development and validation of embedded systems with both safety and security needs. In the case these needs can be demonstrated as independent, the SEISES process consists in starting from the Safety Level and the Security Level(s) resulting from the

corresponding needs and standards, and combine in the most appropriate way the processes, activities etc. required for these levels, with also additional activities to master the possible dependencies (for instance the impact of security means on safety properties, and vice-versa)/

In the important case of "security for safety" it is necessary to investigate in depth the links between the Safety and Security Levels so as to identify which constraints the safety needs may impose on the Security Level(s).

At this stage we propose that the conformity part of the Security Level could be directly related to (and thus derived, defined from) the Safety Level (the topmost part representing the categorisation of the safety needs). Conversely we consider that the effectiveness part should be subject to a dedicated definition activity taking into account an assessment of possible threats and attack scenarios as part of the global risk assessment process. In particular it should be noted that it is not because a system is more critical that it should be provided with stronger security protection mechanisms (because this also depends on its exposure to attacks and their potential (and on the most severe possible impact of these attacks)). On the contrary, it appears obvious that the more the system is critical the more assurance the stakeholder should require on its ability to resist to attacks as intended.

References

[1] "*Information technology – Security techniques – Information security risk management*", ISO/IEC 27005:2011, Second edition, 2011.

[2] P. Bieber, JP. Blanquart, G. Descargues, M. Dulucq, Y. Fourastier, E. Hazane, M. Julien, L. Léonardon, G. Sarouille, "*Security and Safety Assurance for Aerospace Embedded Systems*" in Proceedings of the 6th International Conference on Embedded Real Time Software and Systems (ERTS^2 2012), Toulouse, France, 1-3 February 2012.

[3] "*Certification Considerations for highly integrated or complex aircraft systems*", SAE Aerospace Recommended Practice ARP 4754 and EUROCAE ED79, 1996-11.

[4] "*Space product assurance – Safety*", European Cooperation for Space Standardisation, ECSS-Q-ST-40C, 6 March 2009.

[5] "*Space product assurance – Dependability*", European Cooperation for Space Standardisation, ECSS-Q-ST-30C, 6/3/2009.

[6] JP. Blanquart, JM. Astruc, P. Baufreton, JL. Boulanger, H. Delseny, J. Gassino, G. Ladier, E. Ledinot, M. Leeman, J. Machrouh, P. Quéré, B. Ricque, "*Criticality categories across safety standards in different domains*" in Proceedings of the 6th International Conference on Embedded Real Time Software and Systems (ERTS^2 2012), Toulouse, France, 1-3 February 2012.

[7] *"Software considerations in airborne systems and equipment certification"*, DO-178 issue B, RTCA Inc., and ED-12 issue B, EUROCAE, December 1, 1992.

[8] "*Space product assurance – Software product assurance*", European Cooperation for Space Standardisation, ECSS-Q-ST-80C, 6/32009.

[9] "*ED202 - Airworthiness Security Process Specification*", ED202, EUROCAE WG72, October 2010.

[10] *"Common Criteria for Information Technology Security Evaluation - Part 1: Introduction and general model"*, Version 3.1 Revision 3 Final CCMB-2009-07-001 July 2009