VALU3S

Verification and Validation of Automated Systems' Safety and Security

VALU3S Vocabulary

Document Type	Report
Primary Author(s)	Aleš Smrčka (BUT)
Document Date	2022-06-15
Document Version	2.6 - 2nd approved version
Dissemination Level	Public (PU)
Project Coordinator	Behrooz Sangchoolie, behrooz.sangchoolie@ri.se, RISE Research Institutes of Sweden
Project Homepage	www.valu3s.eu
JU Grant Agreement	876852

ECSEL Joint Undertaking
Electronic Components and Systems for European Leadership

This project has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No 876852. The JU receives support from the European Union's Horizon 2020 research and innovation programme and Austria, Czech Republic, Germany, Ireland, Italy, Portugal, Spain, Sweden, Turkey.

Disclaimer

The views expressed in this document are the sole responsibility of the authors and do not necessarily reflect the views or position of the European Commission. The authors, the VALU3S Consortium, and the ECSEL JU are not responsible for the use which might be made of the information contained in here.

Contributors

Rupert Schlick	AIT	Manuel Schmidt	NXP
Aleš Smrčka	BUT	Ruiz Ricardo	RGB
Thomas Bauer	FRAUNHOFER IESE	Ali Sedaghatbaf	RISE
Hamid Ebadi	INFOTIV	Behrooz Sangchoolie	RISE
David Miguel Ramalho Pereira	ISEP	Pierre Kleberger	RISE
Joseba Andoni Agirre	MGEP	Jose Luis de la Vara	UCLM

Revision History

Version	Date	Author (Affiliation)	Comment
0.1	2020-09-16	Aleš Smrčka (BUT)	Initial Vocabulary Draft
0.2	2020-10-07	Behrooz Sangchoolie (RISE)	Added some comments on version 0.1
0.3	2020-12-03	Aleš Smrčka (BUT)	Added introduction, modified structure
0.4	2021-01-20	Aleš Smrčka (BUT)	Assigned coordinators for new terms
0.5	2021-05-12	Aleš Smrčka (BUT)	Fixed some of the terms
1.0	2021-09-15	Aleš Smrčka (BUT)	Forked to first version (versions marked 1.x)
2.1	2021-11-11	Aleš Smrčka (BUT)	New terms taken from v1.0 review
2.2	2021-11-26	Aleš Smrčka (BUT)	Vocabulary and references pulled from v1.3
2.3	2022-04-13	Aleš Smrčka (BUT)	Updated terms
2.4	2022-05-01	Technical committee	Review of the updated term
2.5	2022-06-15	Aleš Smrčka (BUT)	Second version
2.6	2022-06-16	Behrooz Sangchoolie (RISE)	Second version approval/review

Chapter 1 Introduction

The systems and software engineering disciplines, techniques, and processes have vastly advanced during past decades. The progress is so rapid that terminology is updated in different domains simultaneously, which allows more definitions representing the same thing or unambiguous definitions leading to misunderstanding and faults in designs. Different standards cope with this problem by identifying terms used in the field of information technology and by providing definitions for these terms. Even though such vocabularies and glossaries include clearly defined terms, they either provide several different but similar meanings for different topics or do not include terms that are newly introduced in current research projects. This document was prepared to collect terms used specifically in the context of the VALU3S project and to unify or narrow down their definitions.

The vocabulary is continuously updated during the development of the VALU3S project. This document is the second version consisting of terms with definitions already agreed by the VALU3S consortium.

Chapter 2 Vocabulary

analytical evaluation

methods related to or using analysis or logical reasoning. The analytical evaluation methods as model checks, schedulability analysis, formal model checking can be classified by the formality level as Formal or Semi-formal. Formal methods are based on formal mathematical proofs or correctness and are the most thorough means of V&V. An example of a method classified into analytical, and formal is Model-based Safety Analysis method. Semi-Formal Methods are formalisms and languages that are not considered fully "formal", i.e., not fully mathematically grounded. An example of a method classified into analytical and semiformal is Model-based Threat Analysis method. [1]. *related: experimental evaluation*

artifact

one of many kinds of tangible products consumed or produced during the development or quality assurance of a software product. Examples for artifacts are requirements, architecture design models. source code, binaries, test plans. test cases, test reports [2].

asset

(with regard to security engineering) is anything of value that should be protected from malicious harm [3].

attack

1. an attempt to destroy, expose, alter, disable, steal, or gain unauthorized access to, or make unauthorized use of an asset [4], **2.** a series of steps taken by an attacker to achieve an unauthorized result [5].

authentication

1. the provision of assurance that a claimed characteristic of an entity is correct [4], **2.** the process of verifying a claim that a system entity or system resource has a certain attribute value [6], [7].

automated systems

systems that reduce or eliminate human interventions in processes. In the case of VALU3S, these processes are connected to the project target domains, namely: Automotive, Aerospace, Industrial robotics, Healthcare, Agriculture and Railway.

baseline (of technology or process)

1. snapshot of the state of a service or individual configuration item at a point in time [8]**, 2.** specification or product that has been formally reviewed and agreed upon, that thereafter serves as the basis for further development, and that can be changed only through formal change control procedures [9].

criteria

specific data items identified as contents of information items for appraising a factor in an evaluation, audit, test, or review [10].

critical system
system having the potential for serious impact on the users or environment, due to factors including safety, performance, and security [11].

cybersecurity (and key concepts)
security of cyberspace, where cyberspace itself refers to the set of links and relationships between objects that are accessible through a generalised telecommunications network, and to the set of objects themselves where they present interfaces allowing their remote control, remote access to data, or their participation in control actions within that cyberspace. Cybersecurity shall therefore encompass the CIA paradigm (confidentiality, integrity, and availability) for relationships and objects within cyberspace and extend that same CIA paradigm to address protection of privacy for legal entities (people and corporations), and to address resilience (recovery from attack) [12]. The CIA paradigm refers to the primary cybersecurity attributes. There are also secondary attributes that cybersecurity shall encompass; examples of these attributes are privacy, non-repudiation, authenticity, and freshness [13], [14].

demonstrator
use case, utilising the results achieved by the project [7].

design
1. (process) to define the architecture, system elements, interfaces, and other characteristics of a system or system element, **2.** result of the process as defined in [15], [16].

dimension
distinct components that a multidimensional construct encompasses [17]. In the VALU3S project, the dimensions are used to perform a structured classification of the different V&V methods and tools used in the development of automated systems.

documentation
written or pictorial information describing, defining, specifying, reporting, or certifying activities, requirements, procedures, or results [18].

evaluation
action that assesses the value of something [19]. *related: 1. (noun) analytical evaluation, experimental evaluation, V&V evaluation***; 2.** ***(****adj) evaluation criterion, evaluation scenario*

evaluation criterion
safety, security, privacy, or other criterion to evaluate the effectiveness of the developed processes and methods [7].

evaluation scenario
high-level classification of the underlying test requirements of a Use Case and a statement on "What" needs to be evaluated. The scenarios in VALU3S are a result of interviews with stakeholders within the domains, the vast knowledge of the project partners of their domain, and a close cooperation between partners [20].

error

incorrect state of the system [21]. *related: human error*

experimental evaluation

methods related to or based on experience or experiment. The experimental evaluation methods can be classified into: 1) Testing, 2) Monitoring, and 3) Simulation. [1]. *related: analytical evaluation*

fault

abnormal condition that can cause an element or an item to fail [22], [13].

failure

termination of the ability of an element or an item to perform a function as required [22], [13].

framework

reusable design (models and/or code) that can be refined (specialized) and extended to provide some portion of the overall functionality of many applications [23]. The framework used in VALU3S project permits the classification of the V&V methods and tools as well as Use Case properties into different dimensions. *related: multi-layered / multi-dimensional framework*

formal analysis (=formal analytical evaluation)

process aimed at providing verifiable evidence of the correctness of properties of a system using rigorous formal languages, methods, and models which are mathematically well defined [24]. *related: semi-formal analysis*

functional safety

1. part of the overall safety relating to the EUC (Equipment Under Control) and the EUC control system that depends on the correct functioning of the E/E/PE (Electrical/Electronic/Programmable Electronic) safety-related systems [25], **2.** absence of unreasonable risk due to hazards caused by malfunctioning behaviour of E/E (electrical and/or electronic) systems, where hazard is a potential source of harm caused by malfunctioning behaviour of the item [22].

human error

human action that produces an incorrect result [26]. *related: error*

incident

a group of faults/attacks that can be distinguished from other faults/attacks because of the distinctiveness of the source of faults/attacks, objectives, sites, and timing [5].

layer

partition resulting from the functional division of a software system, where layers are organized in a hierarchy [18]. In VALU3S, a layer is a way of classifying dimensions of the multi-dimensional framework; thus, it does not necessarily refer to functional division of a software system.

model

1. representation of something that suppresses certain aspects of the modelled subject [23], **2.** semantically closed abstraction of a system or a complete description of a system from a particular perspective [18].

monitoring

lightweight and dynamic verification technique that involves observing the internal operations of a system and/or its interactions with other external entities, with the aim of determining whether the system satisfies or violates a correctness specification [1].

multi-layered / multi-dimensional framework

a multi-dimensional framework facilitates the definition, classification, and collection of information. For example, the VALU3S multi-dimensional layered framework offers a design to store information regarding the V&V activities of Safety, Cybersecurity and Privacy properties. [1]. *related: framework*

privacy (and key concepts)

degree to which unauthorized parties are prevented from obtaining "personal" sensitive information [3]. Note that, here “personal” is added to the Firesmith’s definition [3] to explicitly address the concern connected to the personal information. Privacy includes the following subfactors:

- Anonymity is the degree to which the users' identities are prevented from unauthorized storage or disclosure.
- Confidentiality is the degree to which sensitive information is not disclosed to unauthorized parties (e.g., individuals, programs, processes, devices, or other systems).

The above definition is also in line with the definition provided in [27], according to which, privacy is guaranteed if the relation between an entity and a set of information is confidential. Thereby, anonymity is the property that the relation between an entity and its identity is confidential.

requirement

condition or capability that must be met or possessed by a system, system component, product, or service to satisfy an agreement, standard, specification, or other formally imposed documents [28].

safety (and key concepts; some are already included, such as error, fault, failure)

1. freedom from unacceptable risk where risk could be defined as a combination of the probability of occurrence of harm and the severity of that harm [25], **2.** absence of catastrophic consequences on the user(s) and the environment [13].

safety/cybersecurity/privacy requirement

requirement that is needed to ensure the safety/cybersecurity/privacy of the product [29].

semi-formal analysis (=semi-formal analytical evaluation)

process for evaluation of systems and components by using structured means whose application does not result in a mathematical proof [24]. *related: formal analysis*

simulation

approximate imitation of the operation of a system (or process) that represents its operation over time. It is based on the development or use of digital models that behave or operate like real-world systems or components, and on the provision of real-world-like inputs [1], [24].

target domain

targeted industry, where the use cases are selected from. In case of VALU3S, the target domains are Automotive, Aerospace, Industrial robotics, Healthcare, Agriculture and Railway [30].

test case

set of preconditions, inputs (including actions, where applicable), and expected results, developed to drive the execution of a test item to meet test objectives, including correct implementation, error identification, and checking quality and other valued information [31].

test item

1. component of a system under evaluation (either analytical or experimental), **2.** specific artifact of V&V action leading to evaluation of a test case. *related: test case*

testing

quality assurance activity in which systems, subsystems, or components are executed under specified conditions, the results are observed or recorded, and an evaluation is made of some aspect of the system or component [32].

threat

potential cause of an unwanted incident, which can result in harm to a system or organization [4].

tool

a computer program or technical asset that implements or supports an engineering task or parts of it and often supports the automated the execution of this engineering task or parts of it [32].

use case

description of how users will implement and work with the developed solution in their industry specific productive environment. It outlines the solution from a user's point of view. Each use case is represented as a sequence of simple steps, beginning with a user's goal and ending when that goal is fulfilled [20].

user story

simple narrative illustrating the user goals that a software function will satisfy [33], [1].

V&V (Verification and Validation)

process of determining whether the requirements for a system or component are complete and correct, the products of each development phase fulfil the requirements or conditions imposed by the previous phase, and the final system or component complies with specified requirements [18].

V&V evaluation

assessing impact of integrating V&V workflows with V&V methods and tools into the product life-cycle; assessment should be based on selected evaluation criteria [7]. *related: evaluation, evaluation criteria*

V&V method
a particular procedure for V&V, especially a systematic or established one [24].

V&V tool
a computer program or technical asset that implements a V&V method or parts of it and often supports the automated the execution of a V&V method or parts of it [34] (derived from [32]). *related: tool*

V&V workflow
an orchestrated and repeatable pattern of V&V activities that provide services or process information and consists of sequence of operations [2].

validation
1. confirmation, through the provision of objective evidence, that the requirements for a specific intended use or application have been fulfilled [35], [15], [16] **2.** process of providing evidence that the system, software, or hardware and its associated products satisfy requirements allocated to it at the end of each life cycle activity, solve the right problem (e.g., correctly model physical laws, implement business rules, and use the proper system assumptions), and satisfy intended use and user needs [36].

verification
1. confirmation, through the provision of objective evidence, that specified requirements have been fulfilled [9], [35], [15], [16], **2.** process of evaluating a system or component to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase [36].

vulnerability
1. weakness of an asset or control that can be exploited by one or more threats [2], **2.** a weakness in a system allowing unauthorized action [3]. The term can also be used in other topics: **3.** *design v.* a vulnerability inherent in the design or specification of hardware or software whereby even a perfect implementation will result in a vulnerability [3], **4.** *implementation v.* a vulnerability resulting from an error made in the software or hardware implementation of a satisfactory design [3], **5.** *configuration v.* a vulnerability resulting from an error in the configuration of a system [3].

References

[1] MGEP et al., "Deliverable D2.2 - Final multi-dimensional layered framework," VALU3S Consortium, Dec. 31, 2020.

[2] FRAUNHOFER IESE et al., "Deliverable D4.4 - Initial detailed description of improved process," VALU3S Consortium, Oct. 31, 2021.

[3] D. Firesmith, "Common Concepts Underlying Safety, Security, and Survivability Engineering.," Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, CMU/SEI-2003-TN-033., 2003.

[4] "ISO/IEC 27000:2020 — Information technology — Security techniques — Information security management systems — Overview and vocabulary".

[5] J. D. Howard and T. A. Longstaff, "A Common Language for Computer Security Incidents," Sandia National Laboratories, Albuquerque, New Mexico 87185 and Livermore, California 94550, techreport SAND98-8667., 1998.

[6] Network Working Group, "RFC 4949: Internet Security Glossary," https://datatracker.ietf.org/doc/html/rfc4949, 2007.

[7] BUT et al., "Deliverable D5.2 - Final demonstration plan and a list of evaluation criteria," VALU3S Consortium, Oct. 31, 2021.

[8] "ISO/IEC 19770-1:2012 Information technology — Software asset management — Part 1: Processes and tiered assessment of conformance, 3.1".

[9] "ISO/IEC 12207:2008 — Systems and software engineering — Software life cycle processes".

[10] "ISO/IEC/IEEE 15289:2015 — Systems and software engineering — Content of life-cycle information products (documentation)".

[11] "ISO/IEC 29110-2-1:2015 — Software engineering — Lifecycle profiles for Very Small Entities (VSEs) — Part 2-1: Framework and taxonomy".

[12] C. Brookson, S. Cadzow, R. Eckmaier, J. Eschweiler, B. Gerber, A. Guarino, K. Rannenberg, J. Shamah and S. Gorniak, "Definition of Cybersecurity-Gaps and overlaps in standardisation," Heraklion, ENISA. , 2015.

[13] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," In IEEE Transactions on Dependable and Secure Computing, vol. 1, no. 1, pp. 11–33, doi: 10.1109/TDSC.2004.2., 2004.

[14] A. Lautenbach et al., "Deliverable D2.0, Security models," HEAVENS (HEAling Vulnerabilities to ENhance Software Security and Safety). Project deliverable., 2016.

[15] "ISO/IEC TS 24748-1:2016 — Systems and software engineering — Life cycle management — Part 1: Guide for life cycle management".

[16] "ISO/IEC/IEEE 15288:2015 — Systems and software engineering — System life cycle processes".

[17] "ISO/IEC 33003:2015 Information technology — Process assessment — Requirements for process measurement frameworks, 3.6".

[18] "ISO/IEC/IEEE 24765:2017 — Systems and software engineering — Vocabulary".

[19] "ISO/IEC 15414:2015 Information technology — Open distributed processing — Reference model — Enterprise language".

[20] NXP-DE, "Deliverable D1.1 - Description of use cases as well as scenarios," VALU3S Consortium, Aug. 31, 2020.

[21] "ISO/IEC 15026-1:2013 — Systems and software engineering — Systems and software assurance — Part 1: Concepts and vocabulary".

[22] "ISO 26262-1:2018 — Road vehicles — Functional safety — Part 1: Vocabulary".

[23] "IEEE 1320.2-1998 (R2004) IEEE Standard for Conceptual Modeling Language Syntax and Semantics for IDEF1X97 (IDEFobject)".

[24] AIT et al., "D3.1 - V&V methods for SCP evaluation of automated systems," VALU3S Consortium, Dec. 31, 2020.

[25] "ISO/IEC Guide 51:2014 — Safety aspects — Guidelines for their inclusion in standards".

[26] "IEEE 1044-2009 — IEEE Standard Classification for Software Anomalies".

[27] EVITA Project Consortium, "Deliverable D2.3 - Security requirements for automotive on-board networks based on dark-side scenarios," EVITA (E-Safety Vehicle Intrusion Protected Applications), http://www.evita-project.org/., 2009.

[28] "IEEE 730-2014 — IEEE Standard for Software Quality Assurance Processes".

[29] "ISO/IEC TS 15504-10:2011 — Information technology — Process assessment — Part 10: Safety extension".

[30] VALU3S Consortium, "Grant Agreement, Project number 876852 — VALU3S," May 2020.

[31] "ISO 29119-1-2013 — Software testing — Concepts and definitions".

[32] "IEEE Std 610.12-1990, Standard Glossary of Software Engineering Terminology".

[33] "ISO/IEC/IEEE 26515: 2011 — Systems and software engineering — Developing user documentation in an agile environment".

[34] LLSG et al., "D4.5 - Initial implementation of V&V tools," VALU3S Consortium, Oct. 30, 2021.

[35] "ISO/IEC 25000:2014 — Systems and software Engineering — Systems and software product Quality Requirements and Evaluation (SQuaRE) — Guide to SQuaRE".

[36] "IEEE 1012-2012 — IEEE Standard for System and Software Verification and Validation".

[37] "IEC 61508-4 — Functional Safety of Electrical/Electronic/Programmable Electronic Safety-related Systems (E/E/PE, or E/E/PES, Part 4: Definitions".

VALU3S

www.valu3s.eu

ECSEL Joint Undertaking
Electronic Components and Systems for European Leadership

This project has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No 876852. The JU receives support from the European Union’s Horizon 2020 research and innovation programme and Austria, Czech Republic, Germany, Ireland, Italy, Portugal, Spain, Sweden, Turkey.