INTERNATIONAL STANDARD

ISO/IEC 15026

First edition
1998-11-15

Information technology — System and software integrity levels

Technologies de l'information — Niveaux d'intégrité du système et du logiciel

Reference numbe

Contents

ISO/IEC Copyright Office • Case postale 56 • CH-1211 Genève 20 • Switzerland

Printed in Switzerland

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 15026 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software engineering*.

Information technology — System and software integrity levels

1 Scope

This International Standard introduces the concepts of software integrity levels and software integrity requirements. It defines the concepts associated with integrity levels, defines the processes for determining integrity levels and software integrity requirements, and places requirements on each process. This International Standard does not prescribe a specific set of integrity levels or software integrity requirements. These must be established either on a project by project basis, or for a specific sector and/or country. This International Standard is applicable to software only. The system integrity level and the integrity levels of the non-software components are only required in this International Standard to determine the integrity levels of the software components.

This International Standard is intended for use by developers, users, procurers, and assessors of software products or systems containing software for the administrative and technical support of those products and systems.

A software integrity level denotes a range of values of a software property necessary to maintain system risks within tolerable limits. For software that performs a mitigating function, the property is the reliability with which the software must perform the mitigating function. For software whose failure can lead to a system threat, the property is the limit on the frequency or probability of that failure.

Software integrity requirements are requirements that must be met by the software engineering process used to develop the software, requirements that must be met by the software engineering products, and/or requirements that must be true of the software's performance over time in order to provide a degree of confidence in the software that is commensurate with the software's integrity level.

This International Standard does not prescribe the way in which integrity level determination is integrated with the overall system engineering life cycle processes.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this International Standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO/IEC 2382-1:1993, *Information technology - Vocabulary - Part 1: Fundamental terms.*

ISO/IEC 2382-20:1995, *Information technology - Vocabulary - Part 20: System development.*

ISO 8402:1994, *Quality management and quality assurance - Vocabulary.*

IEC 50-191:1990, *International Electrotechnical Vocabulary, Chapter 191: Dependability and quality of service.*

IEC 300-3-9:1995, *Dependability management - Part 3: Application guide - Section 9: Risk analysis of technological systems.*

ISO/IEC 12207:1995, *Information technology - Software life-cycle processes.*

3 Definitions

For the purposes of this International Standard, the definitions given in ISO/IEC 2382-1, ISO/IEC 2382-20, ISO 8402 and IEC 50-191 apply, except as modified or supplemented by the following definitions.

3.1 component: An entity with discrete structure, such as an assembly or software module, within a system considered at a particular level of analysis.

3.2 degree of confidence: In this standard the term degree of confidence is used only to mean the degree of confidence that software conforms to its requirements.

3.3 design authority: The person or organization that is responsible for producing the design of the system.

3.4 failure: The termination of the ability of an item to perform a required function or its inability to perform within previously specified limits.

3.5 fault isolation: The ability of a subsystem to prevent a fault within the subsystem from causing consequential faults in other subsystems.

3.6 function: An aspect of the intended behavior of the system.

3.7 initiating event: An event that can lead to a threat.

3.8 integrity assurance authority: The independent person or organization responsible for assessment of compliance with the integrity requirements.

3.9 integrity level: A denotation of a range of values of a property of an item necessary to maintain system risks within tolerable limits. For items that perform mitigating functions, the property is the reliability with which the item must perform the mitigating function. For items whose failure can lead to a threat, the property is the limit on the frequency of that failure.

3.10 item: An entity such as a part, component, subsystem, equipment or system that can be individually considered. An item may consist of hardware, software or both.

3.11 mitigating function: A mitigating function is a function that, if provided successfully, will prevent an initiating event from becoming a specified threat.

3.12 risk: A function of the probability of occurrence of a given threat and the potential adverse consequences of that threat's occurrence.

3.13 risk dimension: A perspective from which risk assessment is being made for the system (eg safety, economic, security).

3.14 safety: The expectation that a system does not, under defined conditions, lead to a state in which human life, health, property, or the environment is endangered.

3.15 security: The protection of system items from accidental or malicious access, use, modification, destruction, or disclosure.

3.16 software integrity level: The integrity level of a software item.

3.17 subsystem: A subsystem is any system that is part of a larger system.

3.18 system: An integrated composite that consists of one or more of the processes, hardware, software, facilities and people, that provides a capability to satisfy a stated need or objective.

3.19 systematic failure: A failure related in a deterministic way to a certain cause, which can only be eliminated by a modification of the design or of the manufacturing process, operational procedures, documentation or other relevant factors.

3.20 system integrity level: The integrity level of a system.

3.21 threat: A state of the system or system environment which can lead to adverse effect in one or more given risk dimensions.

4 Symbols and abbreviations

There are no symbols used in this International Standard. Where appropriate, abbreviations are expressed fully where appearing in the text for the first time.

5 Software integrity levels framework

5.1 How to use this International Standard

The concept of an independent integrity assurance authority is fundamental to the proper use of this International Standard. The integrity assurance authority is the person or organization responsible for certifying compliance with the integrity requirements. Decisions made during negotiation between the design authority and the integrity assurance authority are documented. The decisions to be negotiated include determination of relevant risk dimensions, the specific integrity levels to be used, the specific criteria for assigning each level, the degree of benefit to be allowed for specific architectural features of the design, and the requirements on the software resulting as a consequence of being assigned a particular integrity level.

The processes described in this International Standard are presented as distinct from the overall system engineering processes, but it is not the intention of this International Standard to prevent them from being integrated with the system engineering processes. Regardless of how the processes are implemented, compliance with this International Standard requires that all requirements within it are met.

Subclause 5.2 provides an overview of the processes for determining integrity levels and software integrity requirements. Clauses 6, 7 and 8 describe the processes in more detail and define the requirements imposed on the processes.

5.2 Overview

Figure 1 shows an overview of the processes required to determine system and software integrity levels, and software integrity requirements. Table 1 lists the inputs and outputs for each of the three main processes of system integrity level determination, software integrity level determination, and software integrity requirements determination.

A risk based approach to integrity level determination is used within this International Standard. Therefore the first step in determining the corresponding system integrity level involves the performance of a risk analysis. IEC Standard 300-3-9 «Risk Analysis of Technological Systems» provides guidance on the performance of risk analysis. Sufficient information must be acquired about the system, its environment and the risk dimensions relevant to the system to allow a risk analysis to be performed. The risks analyzed should cover all relevant risk dimensions such as safety, economic and security as agreed to by the design authority and integrity assurance authority.

Any risks identified by the risk analysis must be evaluated to determine if the risk is tolerable. Once the system design has been analyzed and evaluated to have tolerable risks, a system integrity level is assigned to the system. The system integrity level reflects the worst case risk that is associated with the system.

The integrity level of the software within the system is initially assigned to be the same as the system integrity level. The design of the system may be analyzed to determine if there are architectural features in the system design that would justify assigning a lower integrity level to the software than that of the system.

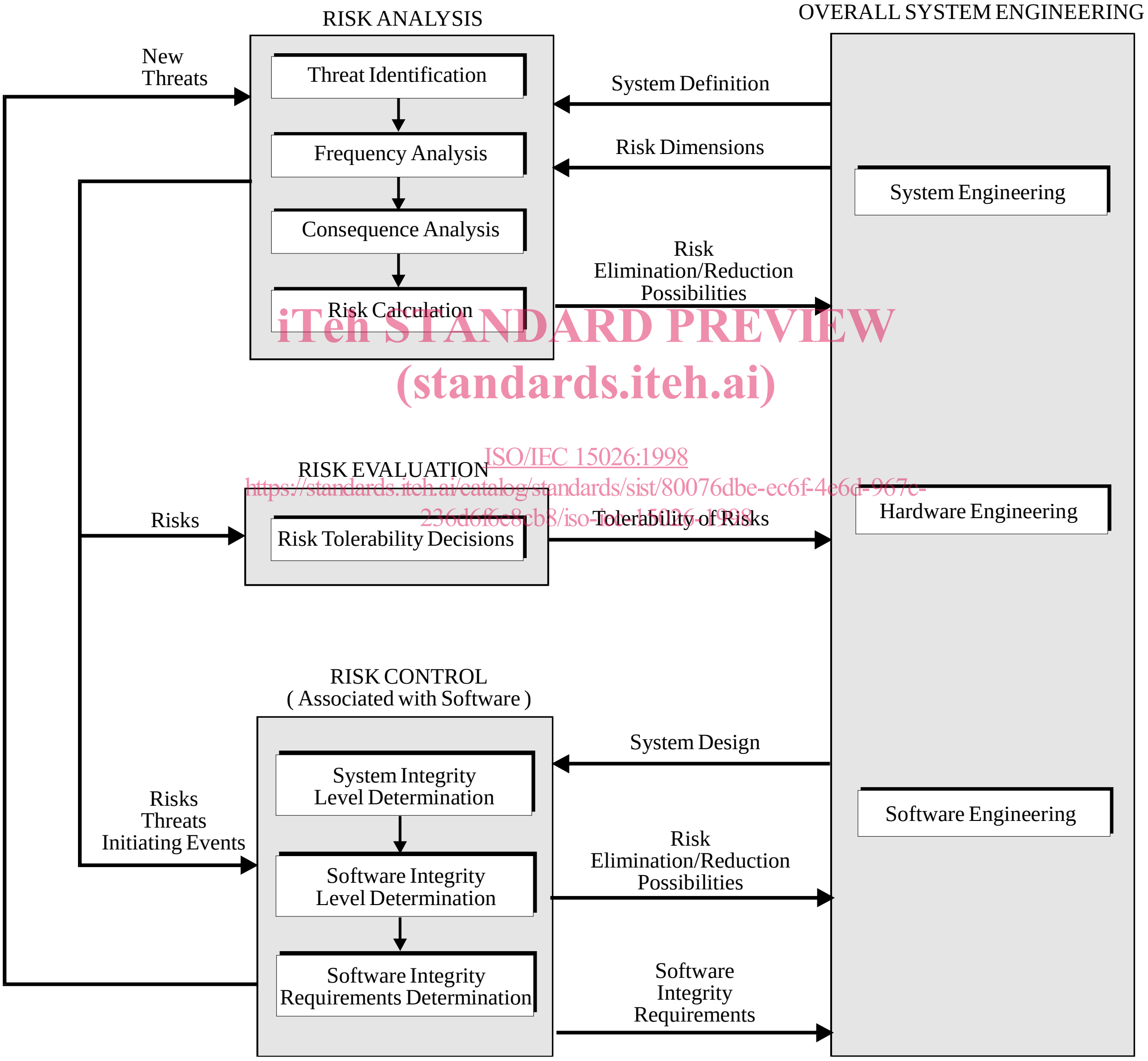
RISK ANALYSIS
OVERALL SYSTEM ENGINEERING
New Threats
Threat Identification
Frequency Analysis
Consequence Analysis
Risk Calculation
System Definition
Risk Dimensions
System Engineering
Risk Elimination/Reduction Possibilities
RISK EVALUATION
Risks
Risk Tolerability Decisions
Tolerability of Risks
Hardware Engineering
RISK CONTROL
(Associated with Software)
System Design
System Integrity Level Determination
Software Integrity Level Determination
Software Integrity Requirements Determination
Risks
Threats
Initiating Events
Risk Elimination/Reduction Possibilities
Software Integrity Requirements
Software Engineering

Figure 1 - Overview of Process for Software Integrity Level Determination and Application

Table 1 - Inputs and Outputs

Process	Inputs	Outputs
System Integrity Level Determination	- relevant risk dimensions - system definition - environment definition - system architecture (if available)	- risks - threats - tolerable frequency/probability of threat occurrence - initiating events - initiating event frequency/probability - system integrity level
Software Integrity Level Determination	- system integrity level - subsystem/software architecture - a list of threats and for each threat: - tolerable frequency or probability of occurrence of the threat - initiating events that may lead to the threat - expected frequency or probability of occurrence of each initiating event	- subsystem/software integrity level - architectural features credited with reducing the integrity level
Software Integrity Requirements Determination	- subsystem/software integrity level	- software integrity requirements

A system is an integrated composite that consists of one or more components. A component can be solely software, solely hardware, or can be a subsystem consisting of a further breakdown of components. Initially, any software component of a system is assigned the integrity level of the system. Software integrity level determination involves the analysis of the system architecture to determine if the integrity level for a subsystem can be lowered from that of the system integrity level. This process can be applied recursively until the integrity level for a subsystem consisting solely of software has been determined, or until the integrity level of the subsystem containing software is acceptable to the design authority and integrity assurance authority for assignment to any software component in the subsystem.

It is possible that during the analysis of the architecture that new threats and risks will be identified that were not determined during previous risk analysis. This would require risk analysis to be repeated to take into account the new risk information.

The software integrity level is an assignment which represents either the degree of reliability of provision of a mitigating function, or the required limit on the frequency of failure that could result in a threat. Since software failures are strictly systematic failures, the software integrity level is an indication of the degree of confidence required that either the mitigating function is provided reliably or the failure that would result in a threat will not occur.

The determination and application of software integrity levels is part of the overall process of risk management. Risk management is conducted throughout a system or software product's life, and may be done iteratively as various levels of detail of the design are decided, or as the design evolves. Figure 1 shows the relationships between the overall system engineering processes and the risk management processes of risk analysis, risk evaluation and risk control.

Iteration can occur if the system design is modified to incorporate risk elimination/reduction possibilities, if new threats are identified as the system and software integrity levels are assigned, or if the system design does not result in a tolerable risk.