前　言

本标准等同采用国际标准 ISO/IEC 15026:1998《信息技术　系统及软件完整性级别》。

本标准定义了与完整性级别相关的概念，定义了确定完整性级别和软件完整性需求的过程，并提出对每个过程的需求。

本标准由中华人民共和国信息产业部提出。

本标准由中国电子技术标准化研究所归口。

本标准由中国电子技术标准化研究所负责起草。

本标准主要起草人：胡九川、罗锋盈、蔡愉祖。

ISO/IEC 前言

ISO(国际标准化组织)和 IEC(国际电工委员会)是世界性的标准化专门机构。国家成员体(它们都是 ISO 或 IEC 的成员国)通过国际组织建立的各个技术委员会参与制定针对特定技术范围的国际标准。ISO 和 IEC 的各技术委员会在共同感兴趣的领域内进行合作。与 ISO 和 IEC 有联系的其他官方和非官方国际组织也可参与国际标准的制定工作。

对于信息技术,ISO 和 IEC 建立了一个联合技术委员会,即 ISO/IEC JTC1。由联合技术委员会提出的国际标准草案需分发给国家成员体进行表决。发布一项国际标准,至少需要 75%的参与表决的国家成员体投票赞成。

国际标准 ISO/IEC 15026 由 ISO/IEC JTC1 信息技术联合技术委员会 SC7 软件工程分技术委员会制定。

中华人民共和国国家标准

信息技术　系统及软件完整性级别

GB/T 18492—2001
idt ISO/IEC 15026:1998

Information technology—System and software integrity levels

1　范围

本标准介绍了软件完整性级别的概念和软件完整性需求，定义了与完整性级别相关的概念，定义了确定完整性级别和软件完整性需求的过程，并提出对每个过程的需求。本标准不规定专门的一组完整性级别或软件完整性需求。它们必须依据某一项目的基础在该项目中加以确定。本标准仅适用于软件。系统完整性级别和非软件部件的完整性级别仅在本标准中被用来确定软件部件的完整性级别。

本标准可供软件产品或包含软件产品的系统的开发者、使用者、采购者和评估人员使用，向他们提供关于这些产品和系统在管理上和技术上的支持。

软件完整性级别表示软件特性的取值范围，该范围对将系统风险保持在可容忍的限度内是必需的。对于执行缓减功能的软件而言，此特性是指软件必须执行缓减功能的可靠性。对于因其失效而导致一个系统威胁的软件而言，此特性是指对该失效的频率或概率的限制。

软件完整性需求是软件开发中软件工程过程所必需满足的需求，是软件工程产品所必需满足的需求；或是为提供与软件完整性级别相适应的软件置信度而对软件在某一时段的性能的需求。

本标准未规定将确定软件完整性级别(的工作)同整个系统工程生存期过程结合在一起的方法。

2　引用标准

下列标准所包含的条文，通过在本标准中引用而构成为本标准的条文。本标准出版时，所示版本均为有效。所有标准都会被修订。使用本标准的各方应探讨使用下列标准最新版本的可能性。

GB/T 5271.1—2000　信息技术　词汇　第1部分：基本术语(eqv ISO/IEC 2382-1:1993)

GB/T 5271.20—1994　信息技术词汇　20部分　系统开发(idt ISO/IEC 2382-20:1990)

GB/T 6583—1994　质量管理和质量保证　术语(idt ISO 8402:1994)

GB/T 8566—2001　信息技术　软件生存周期过程(idt ISO/IEC 12207:1995)

IEC 50-191:1990　国际电工词汇，191章：可信性和服务质量

IEC 300-3-9:1995　可信性管理　第3部分：应用指南　第9章：技术系统的风险分析

3　定义

除下述定义所作的修改或补充外，GB/T 5271.1、GB/T 5271.20、GB/T 6583和IEC 50-191中给出的定义适用于本标准。

3.1　部件　component

在一个特定的分析层次上考虑的系统中带有分立结构的实体。诸如一个组合或软件模块。

3.2　置信度　degree of confidence

在本标准中，置信度仅用于表示软件同其需求相符合的置信度。

3.3　设计机构　design authority

中华人民共和国国家质量监督检验检疫总局2001-11-02批准　　2002-06-01实施

负责产生系统设计的人或组织。

3.4 失效 failure

一个项未能或不能在预先规定的限制内执行某个要求的功能的状况。

3.5 故障隔离 fault isolation

子系统防止它的一个故障引发其他子系统发生相应故障的能力。

3.6 功能 function

系统的预期行为的一个方面。

3.7 引发事件 initiating event

能导致一个威胁的事件。

3.8 完整性保证机构 integrity assurance authority

负责评估完整性需求的符合性的独立的人或组织。

3.9 完整性级别 integrity level

项的某个特性的取值范围的一种表示，该特性取值范围对将系统风险保持在可容忍的限度内是必需的。对于执行缓减功能的项，此特性是指项必须执行缓减功能的可靠性。对于因其失效能导致一个威胁的项，此特性是指对该失效的频率或概率的限制。

3.10 项 item

能够作为单独考虑的一个实体，如一个零件、部件、子系统、设备或系统。一个项可以包括硬件、软件或两者兼而有之。

3.11 缓减功能 mitigating function

缓减功能是这样的功能，若其成功地提供，它将防止引发事件转变成为具体的威胁。

3.12 风险 risk

一个给定威胁发生的概率及该威胁发生后的潜在不利后果的函数。

3.13 风险维 risk dimension

对系统进行风险评估采用的一种视点(如安全性、经济性、安全保密性)。

3.14 安全性 safety

对系统在规定的条件下不导致危害人类生命、健康、财产或环境的一个状态的期望值。

3.15 安全保密性 security

对系统各项的保护，使其免于受到偶然的或恶意的访问、使用、更改、破坏及泄露。

3.16 软件完整性级别 software integrity level

软件项的完整性级别。

3.17 子系统 subsystem

子系统是作为较大系统的一部分的任何系统。

3.18 系统 system

一个集成的复合体，它由一个或多个过程、硬件、软件、设施和人员组成，提供满足明确陈述的要求或目标的能力。

3.19 系统性失效 systematic failure

以确定的方式与某个确凿的原因相关的失效，该失效仅能通过设计、制造过程、操作规程、文档或其他相关因素的更改才能排除。

3.20 系统完整性级别 system integrity level

系统的完整性级别。

3.21 威胁 threat

系统或系统环境的一种状态，它能导致一个或多个给定的风险维内的负面作用。

4 符号和缩略语

本标准中没有使用符号。缩略语在文中第一次出现时用全称表示。

5 软件完整性级别框架

5.1 如何使用本标准

独立的完整性保证机构是正确应用本标准的基础。完整性保证机构是负责验证完整性需求符合性的人或组织。设计机构与完整性保证机构通过协商所作出的决定都要文档备案。需要协商的决策内容包括确定相关风险维、使用的具体的完整性级别、为每个级别规定的明确的标准、对具体的设计结构特征所允许的受益度，以及因软件被分配一个特定的完整性级别而产生的对该软件的需求。

本标准中描述的过程同整体系统工程的过程有区别，但本标准无意防止这些标准同系统工程的过程相集成。无论这些过程如何被实现，它们与本标准的符合性意味着本标准中的所有需求均需予以满足。

5.2 提供了确定完整性级别和软件完整性需求的过程概貌。第 6、7 和 8 章更详细地描述了这些过程并定义对这些过程所提出的需求。

5.2 概貌

图 1 示出了一个决定系统和软件完整性级别，以及软件完整性需求所要求的过程概貌。表 1 列出了确定系统完整性级别、软件完整性级别和软件完整性需求三个主要过程的各自的输入和输出。

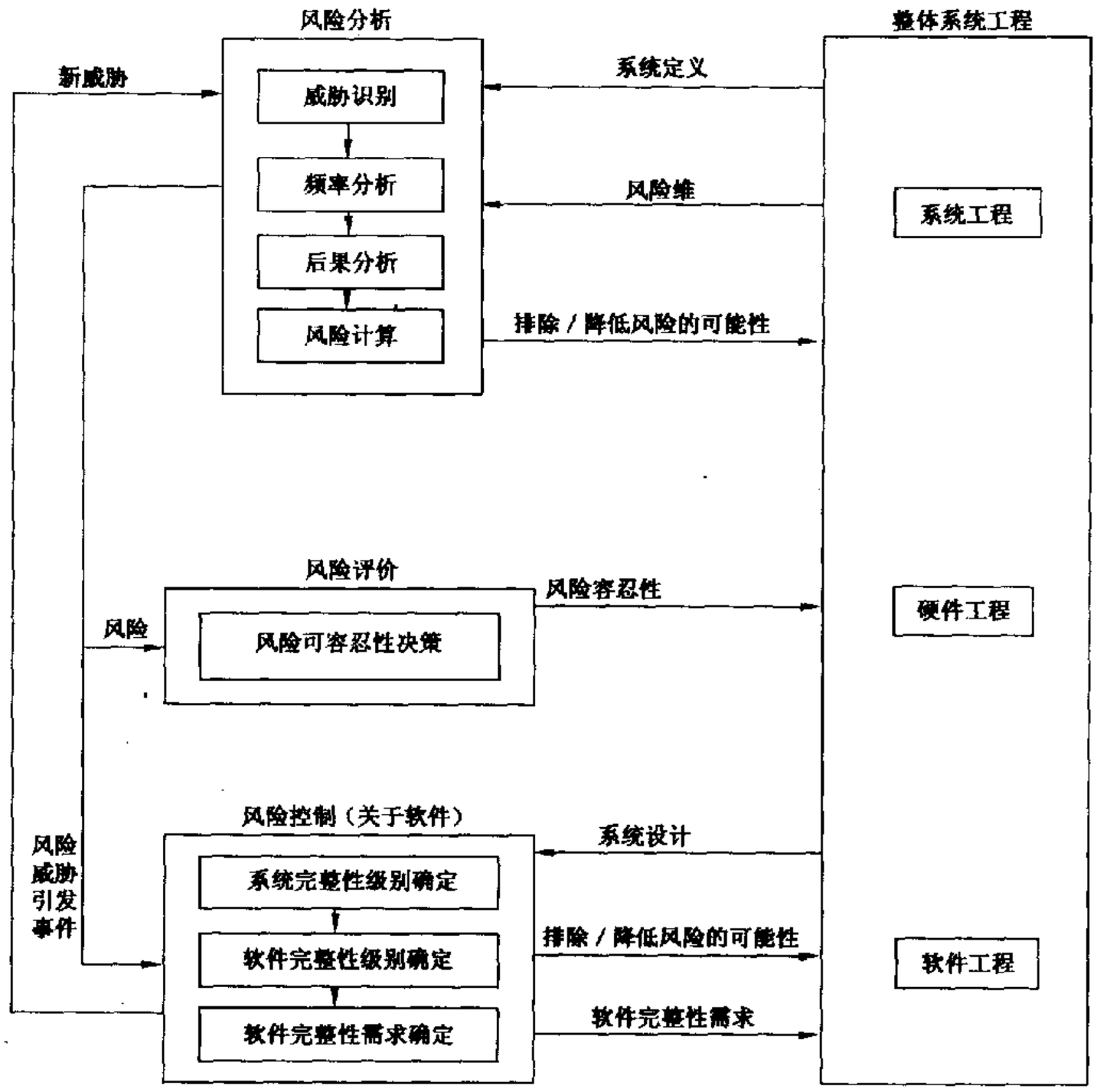

图 1 确定和应用软件完整性级别的过程概貌

本标准中采用了基于风险(分析)的完整性级别的确定方法。所以，确定相应系统完整性级别的第一

步是进行风险分析。IEC 300-3-9 提供了执行风险分析的指南。为了实施风险分析，必须获取关于系统、系统环境、以及与系统相关的风险维方面的足够信息。风险分析应覆盖设计机构与完整性保证机构之间商定的所有相关的诸如安全性、经济性和安全保密性等风险维。

对在风险分析中标识出来的任何风险，必须予以评估以确定该风险是否可以容忍。一旦系统设计经过分析和评价有可容忍的风险，就为系统分配一个系统完整性级别。系统完整性级别反映了系统中存在的在最坏情况下的风险。

系统中软件的完整性级别最初被分配为与系统的完整性级别相同的级别。可以对系统的设计加以分析，以确定系统设计中是否存在为软件分配一个比系统完整性级别低的完整性级别的体系结构特征。

表 1　输入和输出

过　程	输　入	输　出
系统完整性级别确定	——相关风险维 ——系统定义 ——环境定义 ——系统体系结构(若可提供)	——风险 ——威胁 ——可容忍频率/威胁发生的概率 ——引发事件 ——引发事件的频率/概率 ——系统完整性级别
软件完整性级别确定	——系统完整性级别 ——子系统/软件体系结构 ——威胁清单以及对每个威胁而言： ——威胁可容忍频率或威胁发生的概率 ——可能导致威胁发生的引发事件 ——引发事件的期望频率或每个引发事件发生的概率	——子系统/软件完整性级别 ——被确认的降低完整性级别的体系结构特性
软件完整性需求的确定	——子系统/软件完整性级别	——软件完整性需求

系统是由一个或多个部件组合集成的。一个部件可以是单独的软件、单独的硬件或由分解为更细的部件组成的子系统。最初，系统的完整性级别将分配给系统的任何软件部件。确定软件完整性级别涉及实施对系统体系结构的分析，以便确定子系统的完整性级别能否低于系统的完整性级别。这样的实施过程可以循环地进行，直至仅包含软件的子系统的完整性级别得以确定；或者包含软件的子系统的完整性级别被设计机构和完整性保证机构所认可，该认可的级别可以分配给子系统中的任何软件部件。

很可能在系统的体系结构分析的过程中，将发现在先前的风险分析中未曾遇到的新的威胁和风险。因此，有必要将新的风险信息加以考虑而重新实施风险分析。

软件完整性级别或者是表示提供缓减功能的可靠程度的分配，或者是表示对可能导致威胁产生的失效频率的限制的分配。由于软件失效是严重的系统性失效，所以软件完整性级别表示一种置信度指标，其表示可靠地提供缓减功能的真实程度，或者表示不导致某威胁产生失效的真实程度。

软件完整性级别的确定和应用是整个风险分析过程的一部分。在系统和软件产品的生存周期内实施风险管理，而且可能随不同层次的设计细节的确定或随设计的改进而反复进行。图 1 示出了整个系统工程过程与风险分析、风险评价和风险控制等风险管理过程之间的相互关系。

如果系统设计的改进可消除或降低风险，或者当系统和软件完整性级别被分配而发现新的威胁或者系统设计不能具有可容忍的风险，那么风险管理将重复进行。

软件完整性级别用来确定如何控制风险。相对于软件部件而言，就是明确哪些在软件和软件工程方面的需求，从而获得在系统风险中起相应作用的软件的置信级别。这些需求就是软件完整性需求。

6 系统完整性级别的确定

系统完整性级别对应于与系统相关的风险可容忍性级别。由于系统的失效可导致一个威胁产生、或者在系统所包含的在其环境中缓减引发事件后果的功能可能导致一个威胁产生，故系统与风险相关联。确定系统完整性级别的步骤如下：

a）实施风险分析，确定系统相关的风险级别。风险分析是利用可供采用的信息发现的威胁，并且估计与这些威胁相关的风险的过程。

b）风险评估是判断风险的可容忍度的过程。风险分析与风险评估的结论可能导致对系统设计的修改以消除或降低风险，进而可能需要重复进行风险分析和评估。

c）只要具有可容忍风险的系统设计被确认，系统的完整性级别即可确定。完整性级别的准确级别数、区别不同级别的标准均由设计机构和完整性保证机构协商确定。

6.1 风险分析

实施风险分析要回答如下三个基本问题：

什么可能导致错误？（通过威胁识别）

发生的可能性如何？（通过频率分析）

后果是什么？（通过后果分析）

利用这些分析的结果，可以计算风险。对于每个识别出的风险，其风险分析的输出为：

a）用合适的术语表述风险；

b）与风险相关的威胁；

c）与每个威胁相关的引发事件；以及

d）引发事件发生的假设频率。

经过风险分析而得出的风险表述可用于风险评估过程以确定风险是否可以容忍。在系统和软件完整性级别的确定过程中，风险分析的所有结果，可用于确定对系统中软件部件所需求的完整性级别。

风险分析的适用指南是 IEC 300-3-9《技术系统的风险分析》。由于在 IEC 300-3-9 中用到安全性方面的专门术语，术语“危害(hazard)”和“伤害(harm)”必须分别被解释成“威胁”和“负面作用”。

风险分析可以覆盖安全性、经济性和保密性等诸多风险维。要覆盖的具体的风险维应由设计机构和完整性保证机构确定。

6.1.1 威胁识别

应把与系统相关的威胁同可能导致威胁的引发事件一起识别。若系统失效，按规定的系统操作或者系统在其环境中执行缓减引发事件后果的功能可导致威胁的产生，则这些威胁同系统相关联。发现先前未曾识别出来的威胁，将采用适于具体情况的方法，并将方法记录备案(见 IEC 300-3-9)。在识别威胁过程中应考虑系统体系结构(若可提供)，以确保系统中所使用技术的特定的失效模式也得到考虑。

6.1.2 频率分析

频率分析用来估计经威胁识别而确定的每个引发事件发生的可能性。当系统的一个失效是一个引发事件，频率分析不用来估计失效发生的可能性，而是用来确定与可容忍风险目标相一致，并且是可实际达到的失效的频率的限度。

用相关的历史数据估计事件频率，用诸如故障树或事件树分析(IEC 300-3-9)等技术来综合它，或用专家判断来估计事件频率。

在系统完整性级别的确定过程中实施频率分析，系统将视作一个黑盒子，系统的体系结构亦将不予以考虑。系统的体系结构将在软件完整性级别的确定过程中考虑。经频率分析而得出的事件频率可以用定量的词语表述，或者用定性的词语表述频率的范围(如频繁的、可能的、偶然的、间接性的、不可能的或难以置信的)。

频率分析中，要考虑到某些引发事件与其他事件相结合而导致威胁的产生，或者作为一个事件后果

的部分情形。

6.1.3 后果分析

如果引发事件发生，后果分析用来估计威胁的严重性。要明确指出可缓减引发事件后果的任何措施。每一个威胁归入为一个后果类，这些后果类描述了后果的不同程度的严重性(如灾难性、重大性、严重性、次要性)。

6.1.4 风险计算

与每个威胁相关的风险，可用风险矩阵来计算。风险矩阵将引发事件发生的频率同引发事件产生的后果的严重性联系了起来。在风险矩阵中，对每一个频率与严重性的组合配对指定一个风险类(如高风险、中等风险、低风险或微小风险)。表 2 为 IEC 300-3-9 中一个风险矩阵的例子。

表 2 风险矩阵的例子

发生频率	频率(每年)	后果的严重性			
		灾难性	重大性	严重性	次要性
频繁性	>1	高	高	高	中等
可能性	$1\sim10^{-1}$	高	高	中等	低
偶然性	$10^{-1}\sim10^{-2}$	高	高	低	低
间接性	$10^{-2}\sim10^{-4}$	高	高	低	低
不可能性	$10^{-4}\sim10^{-6}$	高	中等	低	微小
难以置信	$<10^{-6}$	中等	中等	微小	微小

设计机构和完整性保证机构需对所采纳的计算风险的风险矩阵达成一致意见。达成对风险矩阵的一致性意见需要对引发事件发生频率的适当范围、后果类及其定义、与每一个频率范围和后果类相关的风险类。

6.2 风险评估

设计机构和完整性保证机构要对通过风险分析而得出的风险的可容忍性进行裁决。在某些地区，这样裁决要依据该地区立法机构所建立的法规来进行。

6.3 系统完整性级别的确定

最初分配给系统的完整性级别同赋给与系统相关的威胁最高风险类相对应。完整性级别数同已确定的风险类的数量相对应。表 3 给出风险等级到系统完整性级别的映射例子。

表 3 风险等级映射到系统完整性级别

风险等级	系统完整性级别
高	A
中等	B
低	C
微小	D

本标准没有明确指出完整性级别的级别数或完整性级别的记法。

7 软件完整性级别的确定

软件完整性级别是系统完整性级别在包含软件部件或仅包含软件部件或仅包含软件的子系统上的分配。一个子系统的软件完整性级别最初与系统的完整性级别相同。

除非因进行以下四个方面的考虑而使软件完整性级别相应降低，子系统的软件完整性级别与系统的完整性级别相同：

a）考虑系统的缓减子系统失效后果的体系结构特征。不考虑将导致一个系统威胁产生；

b）考虑系统的提供冗余缓减功能体系结构特征，这些特征使引发事件被缓减，尽管有单个或多个子系统的失效已经分配了缓减功能；

c）考虑子系统在识别引发事件或缓减功能方面的作用。

7.1 软件完整性级别确定前提

a）系统被赋予了一个完整性级别；

b）系统的体系结构特征被充分地、详细地予以定义，从而能够识别子系统的作用和它们的接口；

c）确定完整性级别的过程输入：

——系统的完整性级别

—— 关于威胁的清单，并且对一个威胁要提供以下信息：

——可能导致威胁产生的引发事件

——每个引发事件产生的设计频率或概率

——系统体系结构的充分的、详细的定义。该定义便于确定每个子系统的作用和识别系统体系结构中具有缓减作用的特征；

d）输出为软件完整性级别。

7.2 软件完整性级别的降低

确定一个可能降低的软件完整性级别，必须执行下列步骤：

a）识别出构成整个系统的子系统的集合；

b）确定任何子系统是否孤立地或与其他子系统状态相结合地导致威胁产生。若子系统孤立地失效导致产生一个威胁，则赋予子系统与系统的完整性级别相同的完整性级别。若子系统与其他子系统状态相结合地失效而导致一个威胁，则根据 7.3 中定义的评估过程而得到的结果，子系统的完整性级别可能降低。7.3 中定义的评估过程是非强制性的，只有需要一个较低的子系统完整性级别才实施该过程；

c）确定任何子系统是否孤立地或与其他子系统状态相结合地失效将导致不能提供某个缓减功能。若子系统孤立地失效导致不能提供缓减功能，则赋予子系统与系统的完整性级别相同的完整性级别。若子系统与其他子系统状态相结合地失效而导致不能提供缓减功能，则根据 7.4 中定义的评估过程而得到的结果，子系统的完整性级别可能降低。7.4 中定义的评估过程是非强制性的，只有需要一个较低的子系统完整性级别才实施该过程；

d）确定是否有包含软件部件的子系统，这样的软件的失效不能导致产生威胁，并且软件的功能与任何系统事件的缓减无关。任何这样的软件应赋予最低的完整性级别。必须实施故障隔离以确保软件失效不导致产生威胁。设计机构和完整性保证机构必须在足够的体系结构特征方面取得一致意见，以确保充分的故障隔离。如果通过失效处理机制达到了故障隔离，应分配给该机制有与系统完整性级别相同的软件完整性级别。

从指定的系统完整性级别降低软件的完整性级别，体系结构特征所带来的收益程度，必须由设计机构和完整性保证机构予以规定并认可。

重复使用上述包含步骤 a)至步骤 d)的过程，直到所有仅包含软件的子系统的完整性级别被设计机构和完整性保证机构所接受，并将其赋予这些子系统的任何软件部件。

7.3 降低因其失效可能导致一个威胁的软件完整性级别

对于那些因其失效而导致产生威胁的系统，其完整性级别部分取决于对同系统可容忍的风险目标相一致的系统失效频率的限制。若仅在某一特别状态下软件与其他子系统相结合而失效，导致威胁产生，则可以对软件失效频率的限制适当放宽。频率分析可以用于决定对软件失效频率的最宽松的限制，这种最宽松的限制同可容忍的风险目标相一致，同时，频率分析要将子系统之间的依赖关系加以考虑。确定较紧的对软件失效频率的限制，以进行重复风险计算，进而明确在风险等级和由此通常的在很低级别上的软件完整性级别是否有任何变化。

失效处理机制可用于检测软件失效并采取行动防止软件失效变为引发事件。在这些情况下，即使失效发生和失效处理机制无效，软件失效仅能成为一个引发事件。失效处理机制的例子如下：

a）数据完整性检查（软件机制）；

b）硬件看门狗计时器（硬件机制）；

c）人工恢复（人工机制）。

只要冗余子系统之间的通用模式失效可以避免，冗余能阻止失效导致威胁产生。当软件多样性用于缓减因对软件失效频率的限制而造成的压力，软件多样性带来的受益程度，以及对充分多样性的规定必须受到设计机构和完整性保证机构的认可。

7.4 降低因其失效可能导致不能提供缓减功能的软件完整性级别

若子系统与其他子系统状态相结合的失效仅导致不提供减缓功能，则软件的可靠性会低于系统对缓减功能的要求。正如 7.3 中所指出，用于频率分析的技术可用于确定软件可靠性的系统所要求的可靠性的降低程度。

8 软件完整性需求确定

8.1 置信度

软件完整性级别或者是表示提供缓减功能的所要求的可靠性程度的分配，或者是对可能导致威胁产生的所要求的失效频率的限制的分配。由于软件失效严格说是系统性失效的函数，所以软件完整性级别表示置信度指标，其表示可靠地提供缓减功能的真实程度，或者表示不导致某威胁产生失效的真实程度。软件及其生存周期过程所要求的需求也称作软件完整性需求，如果被满足，将提供必需的置信度。

在软件中获取置信的一些策略如下：

a）采用一些技术，使错误的引入最小化；

b）采用/应用验证和确认技术，把错误的检测最大化；

c）通过操作历史表明/证明软件可靠地提供缓减功能，或失效率低于规定的限制。

8.2 在软件中获得置信度的方法

表 4 简要地给出了一些软件工程活动、活动输出的属性和可用来获得不同的置信度的实现这些属性的方法的例子。本标准不打算规定采用某个特定的软件生存期。GB/T 8566 中描述的其他生存周期过程同样适用。

若软件已存在和已运行了一段时间，软件的置信可通过评价它的运行历史得到。运行历史提供的置信度依赖于各种因素，如软件的复杂性、成功操作的历史和使用软件的方法同按规程使用软件的方法之间的相似性等。

8.3 软件置信度与完整性级别的联系

本标准未规定为了得到适合于给定的软件完整性级别的置信度所要满足的特殊需求。

对于为达到每个完整性级别软件所必需的置信度所要满足的需求，设计机构和完整性保证机构应就其达成协议。

表 4 过程活动、属性和置信度例子

活　　动	属　　性	对于每一完整性级别中获得置信度的方法
软件需求分析	精确性、完整性 正确性、抽象性 一致性、可验证性	1. 结构化方法 2. (1)加上半形式化记法 3. (1)加上形式化记法 4. (3)加上形式化证明

表 4(完)

活　　动	属　　性	对于每一完整性级别中获得置信度的方法
软件设计	需求分析的可追踪性、可验证性、模块性、抽象性	1. 结构化方法 2. (1)加上半形式化记法 3. (1)加上形式化记法 4. (3)加上形式化证明
软件编码	设计的可追踪性、程序结构的无歧义性、程序设计语言的标准化、可维护性	1. 结构编程 2. (1)加上强类型语言 3. (1)加上对语言安全子集的限制 4. (3)加上形式化证明