ICS 35.020
I651

T/ SIA

中 国 软 件 行 业 协 会 团 体 标 准

T/SIA 008.1—2018

就绪可用软件产品（RUSP）安全质量评价标准 第 1 部分　安全质量模型

Ready to Use Software Product (RUSP) Security Evaluation

Criteria—Part 1：Security Quality Model

2018-12-12 发布　　2018-12-12 实施

中 国 软 件 行 业 协 会　发 布

目　录

前　言

本标准按照GB/T 1.1-2009给出的规则起草。

本标准由中国软件行业协会提出并归口。

本标准起草单位：中国软件行业协会系统安全工程分会、大连理工大学、北京天融信网络安全技术有限公司、联想集团（北京公司）、远光软件股份有限公司、北京圣世信通科技发展有限公司。

本标准主要起草人：宋明秋、付晓宇、陈宝国、张然、陈兴跃、李汝鑫、李锋、李广志。

本标准为首次制定。

1 范围

本标准的本部分定义了：

a）软件产品安全质量模型，该模型由9个属性组成，每一个属性又可以进一步细分为一些子属性，这些属性关系到软件的静态安全性和系统的动态安全性。这一模型用于指导就绪可用软件产品的安全质量要求。

b）本部分为就绪可用软件产品（RUSP）安全质量评价标准的第1部分。

c）本部分在GB/T 25000.10/ISO/IEC 25010的系统与软件质量模型基础之上，进一步清晰了软件产品在安全性方面应该具有的属性特征，以及描述这些安全属性的子属性。这些属性是基于网络与信息系统安全的基本特性而建立或划分的，一般认为它们具有原子性特征。

d）本部分可用于指导RUSP软件开发的安全质量管理。

注1：RUSP是指一种打包出售给对其特征和质量没有任何影响的需方的软件产品。典型的情况是，这种软件产品与其用户文档集一起预先包装好出售，或者从Web商店下载。用户能在任何时间通过云计算平台下载使用的软件产品都可以认为是RUSP软件产品。

注2：RUSP的例子包括但不限于：文本处理程序、电子表格、数据库管理软件、图形包、以及用于技术的、科学的或实时的嵌入式功能的软件（例如实时操作系统）、人力资源管理软件、营销管理、生产管理、库存管理、智能手机应用APP、免费软件、以及诸如Web网站和主页生成器之类的Web软件。

2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注明日期的引用文件，仅注日期的版本同样适用于本文件。凡是不注日期文件的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 25000.10-2016 系统与软件工程 系统与软件质量要求和评价（SQuaRE）第10部分：系统与软件质量模型。

GB/T 25000.51-2016 系统与软件质量要求和评价（SQuaRE）第51部分：就绪可用软件产品（RUSP）的质量要求和测试细则。

本标准为首次制定。

3 术语、定义和缩略语

3.1

软件质量 Software Quality

在规定条件下使用时，软件产品满足明确和隐含要求的能力。

3.2

软件安全 Software Security

在面临蓄意威胁其可靠性的事件的情形下依然能够提供所需功能的能力。

3.3

软件安全质量 Software Security Quality

描述了软件产品在受到它方信息安全攻击破坏的时候保护自身及所承载文件、数据的能力，即软件产品内在的本质安全特性。

3.4

本质安全 Intrinsic Security

指通过设计等手段使生产设备或生产系统本身具有安全性，即使在误操作或发生故障的情况下也不会造成事故的功能。具体包括失误—安全功能（误操作不会导致事故发生或自动阻止误操作），以及故障—安全功能（设备、工艺发生故障时还能暂时正常工作或自动转变为安全状态）。

3.5

结构安全 Secure Structures

指软件/系统的框架结构上的安全程度，包括模块组成、块间关联及交互机制的安全性。

3.6

安全属性 Secure Attributes

安全属性是指软件产品本身固有的持久的安全特性，用于描述软件产品的安全质量。

3.7

就绪可用软件产品 Ready to Use Software Product(RUSP)

无论是否付费，任何用户可以不经历开发活动就能获得的软件产品。

注1：RUSP包括：产品说明（包括全部封面信息、数据表、网页信息等）；用户文档集（安装和使用软件所必需的文档），包括为运行该软件产品所要求的操作系统或目标计算机的任何配置；计算机媒体（磁盘、CD-ROM、网络可下载的媒体等）上的软件。

注2：软件主要由程序和数据组成。

注3：本定义也适用于产品说明、用户文档集，以及作为单独的制成品而被生产和支撑的软件，该软件不收取通常的商业费用和证书费用。

4 安全质量模型框架

4.1 模型结构

本标准采用ISO/IEC 25000系列标准中质量模型的树形结构予以表达。其中安全质量用安全属性集合来描述，并在某些情况下将安全属性进一步分解为子属性（某些子属性又被分解为子子属性）。这一层次化分解提供了便利的产品质量分级方法，也与系统工程的结构化分析方法相符合。

4.2 安全质量模型及属性说明

软件安全质量模型包括保密性、完整性、可用性、真实性、访问授权、可记账性/可审

计性、结构安全性、脆弱性和隐私安全性共9个属性特征，这些属性又被进一步分解为下一级的子属性。分解的目的是为了获得对该属性的可导出的质量测度。

4.2.1 安全质量模型

软件安全质量模型框架如图 1 所示。

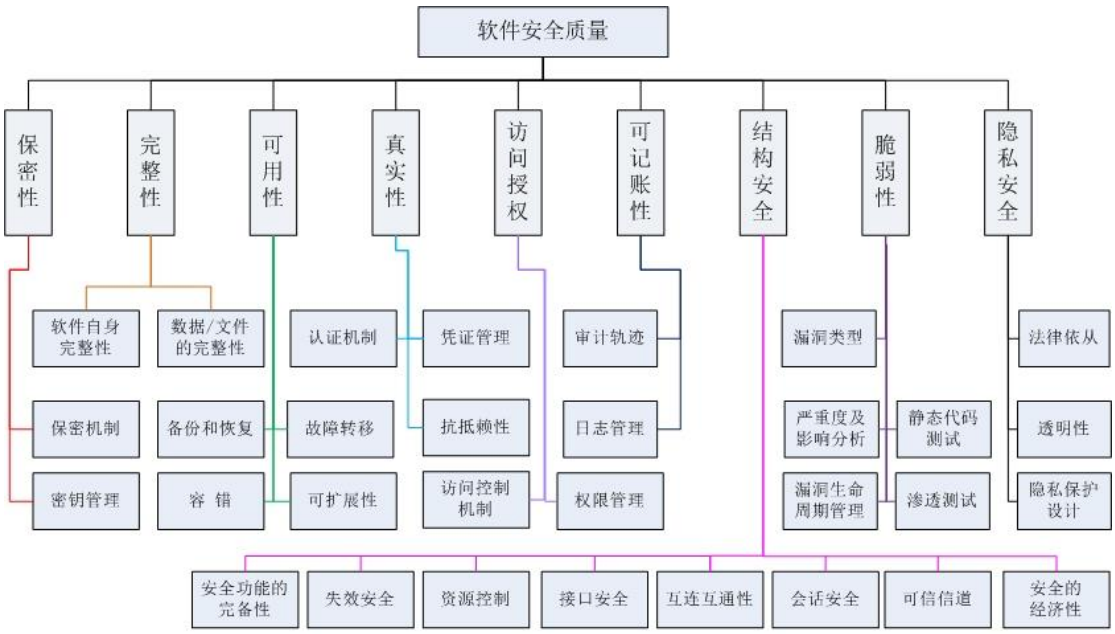

图1 软件安全质量模型框架

4.2.2 属性说明

4.2.2.1 保密性

确保数据只有在授权时才能被访问的程度。主要包括：

a）保密机制。为保护数据资产不被未授权用户非法读取而采取的加密技术或措施，如采用对称加密、非对称加密、信息摘要技术或者隐写术等。

b）密钥管理。对密钥生命周期全过程的安全管理，包括密钥的产生、交换、存储、更新、归档和销毁的过程管理。

4.2.2.2 完整性

描述软件、系统或组件防止未授权访问、篡改计算机程序或数据的程度。主要包括：

a）软件自身的完整性。指软件能够按照预期的功能运行，不受任何有意的或者无意的非法错误所破坏的属性。

b）数据/文件的完整性。指软件在传输、存储信息或数据的过程中，确保所处理的信息或数据不被未授权的人员篡改或在篡改后能够被迅速发现。

4.2.2.3 可用性

描述软件被授权实体访问并按需求使用的特性，即当需要时能存取所需的信息，而未授权的用户则无法获取信息或资源。主要包括：

a）备份和恢复。软件应提供对其自身以及所处理的数据具有备份和复制的能力。

b）容错。即便发生了确定的失效，软件也能继续发挥应有的能力，维持正常运转。

c）故障转移。当服务或数据发生不可用的意外状态时，软件应为用户提供从一个活跃的交易软件、服务器系统、硬件或网络自动或手动转移/切换至安全的备用（冗余）系统的能力。

d）可扩展性。软件应该提供可扩展能力，以满足用户业务升级需要。

4.2.2.4 真实性

描述对象或资源的身份标识能够被证实符合其声明的程度。主要包括：

a）认证机制。确认和识别一个主体或资源所声称的身份的方法和过程，被认证的主体可以是用户、进程、系统和信息等。认证机制可以是基于所拥有的知识（如口令等）、所有权（如智能卡、动态令牌卡等）、生物特征（如人脸识别、指纹识别等）。认证机制可以分为单因素认证、双因素认证以及多因素认证。

b）凭证管理。指凭证的产生、存储、同步、重置和撤销的过程的安全性。

c）抗抵赖性。软件应确保信息的发送方/接收方不能否认其曾经发送/接收过信息。

4.2.2.5 访问授权

根据访问主体的身份和职能为其分配一定的权限，访问主体只能在权限范围内合法访问。主要包括：

a）访问控制机制。指在访问主体与访问对象之间介入的一种安全机制，是对主体访问客体的权限或能力的限制。

b） 权限管理。在身份认证后，对于谁有资格（授权或允许）执行什么操作的权限控制，包括权限授予、验证、更新和回收。

4.2.2.6 可记账性/可审计性

确保一个实体的访问动作可以唯一地被区别、跟踪和记录的特性，以用于信息系统的审计。例如：设置日志记录。

4.2.2.7 结构安全性

描述与业务流程密切相关的安全功能模块的完备性、模块之间接口的安全性、以及互联互通的安全性。主要包括：

a）安全功能的完备性。指针对软件说明书中阐述的安全目标，所实现的安全功能的完备程度。

b）失效安全。即使软件在发生故障的情况下也不会造成损失或者尽量减少损失的能力。

c）资源控制。软件对使用到的资源应保证其安全性，例如：重要资源在使用之前要进行正确的初始化设置，资源分配数量、使用权限和有效时间限制，资源使用过后的及时释放，以及对外部资源的完整性和真实性检查，等等。

d）接口安全。指用户和软件/系统或者软件的组件之间通信时，要保证接口的安全性。例如：屏幕上显示敏感信息时应该采用屏蔽技术，避免信息泄露。

e）互联互通性。指软件上下游之间通信的安全性。例如需要在应用程序之间共享密钥时，如果一个上游应用程序使用特定的密钥对数据进行加密，必须有一个安全的方式将密钥传送给下游的应用程序，以解密数据。

f）会话安全。适用时，软件应限制一个用户可以为某个会话选择的会话安全属性范围。

g）可信信道。软件应为用户和其他可信IT产品之间提供一个可信的信道，以实现对关键安全操作或重要传输数据的保护。

h）安全的经济性。指以较低的开发成本和资源消耗获得具有较高安全质量的软件产品和系统保障。

4.2.2.8 脆弱性

指软件产品本身存在的安全漏洞或Bug，以及这些漏洞被攻击的概率和可能产生的影响。主要包括：

a）漏洞类型。指安全漏洞分类归属的信息。

b）严重度及影响分析。严重度：指根据GB/T 30279-2013漏洞等级划分指南所确定的漏洞的安全级别。影响分析：指利用安全漏洞对目标系统造成的损害程度，可以用对基本安全属性的影响程度来描述，如对保密性、完整性等的影响。

c）漏洞生命周期管理。针对漏洞的发现、利用、修复、公开等漏洞生命周期所进行的预防、收集、消减和发布等一系列管理活动。

d）软件安全测试。软件在正式安装到用户系统之前，应该进行安全测试，主要包括：静态代码测试和渗透测试。

4.2.2.9 隐私安全性

描述软件运行过程中保证个人信息与隐私安全的特性，主要包括：

a）法律依从。指软件在处理个人信息和隐私过程中应遵从国家《网络安全法》的相关内容。

b）透明性。指个人信息收集、使用、传输过程中，所实现的就是所声称的功能。

c）隐私保护设计。为保护用户个人信息和隐私，需要在软件设计中考虑《网络安全法》等法律法规的相关内容，为用户提供安全的隐私保护技术，以防止其隐私信息被其他用户发现并滥用。

注：这里的《网络安全法》是指《中华人民共和国网络安全法》。

4.3 安全质量模型目标

通过软件安全质量模型，确定RUSP软件产品安全质量级别，可以帮助用户快速简便地判断软件的受信任程度。

4.4 利益相关方

软件安全质量模型为各利益相关方对RUSP的安全要求提供了一个框架。利益相关方包括以下类型的用户：

a）主要用户。为了达到主要业务目标与软件进行交互的单位/人员。

b）开发方。依据法律法规及用户需求开发软件产品的单位/人员。

c）运维方。为维护软件的正常运行而提供相关服务的单位/人员。

d）评估者。接受软件的输出结果，但不与软件直接进行业务交互的单位/人员，包括政府和行业主管部门、第三方审计部门等。

4.5 安全质量模型的使用

a）定义的软件产品安全质量模型可以作为一个检查表使用，以便确保软件产品的开发安全质量需求，为软件/系统开发期间所需要考虑的安全要素和安全活动奠定基础。在确定或评价一个计算机系统的安全质量时，可以将本产品质量模型的属性，作为一个集合予以使用。将模型作为需求分解的一部分使用之前，可依据利益相关方的软件产品安全目标和安全策略，对模型进行裁剪，以便标识最重要的、需要在软件开发过程中分配资源予以实现的那些属性和子属性。

b）在特定的使用环境下，不同用户均对使用质量和产品质量有一定的要求，表1给出了不同用户对安全质量特性需求的一些示例。在软件开发或获取之前，宜从利益相关方的视角定义质量需求，分析使用需求，由此产生一个RUSP产品达到使用需求所需要的安全导出功能和安全质量需求。

表1 RUSP软件产品安全质量的用户需求示例

用户需求	主要用户	开发方	运维方	评估者
	交互	开发	维护或移植	使用输出
保密性	信息保密需求	保密机制设计实现	密钥管理维护	质量级别
完整性	完整性需求	完整性机制设计与实现	完整性运维保证	质量级别
可用性	可用性需求	可用性机制实现	可用性运维保证	质量级别
真实性	认证需求	认证机制实现	凭证管理维护	质量级别
访问授权	访问授权需求	控制机制实现	权限管理	质量级别
可记账/审计性	访问日志	日志机制实现	日志管理	质量级别
结构安全性	结构安全需求	安全设计与开发	安全运维	质量级别
脆弱性	漏洞管理需求	保证开发过程安全减少漏洞	漏洞管理	质量级别
隐私安全性	隐私保护需求	隐私保护设计与实现	隐私安全管理	符合性及质量级别

注 1：符合性是指软件产品属性满足安全属性目标的达成程度，用安全质量水平来标识。

注 2：符合性也包括对于国家法律法规的遵从。

5 参考文献

GB/T 25000.1-2010,《软件工程 软件产品质量要求与评价（SQuaRE）：SQuaRE 指南》.

GB/T 19000，《质量管理体系 基础和术语》.

宋明秋.《软件安全开发—属性驱动模式》. 电子工业出版社(北京). 2016.5.