DOI: 10.13656/j.cnki.gxkx.2001.01.006
广西科学 Guangxi Sciences 2001, 8 (1): 15~18

通过 CC标准的思想确定 RS-Linux的安全可信度*
Determining Confidence in Security of RS-Linux Through Concepts of the Common Criteria

石文昌 孙玉芳
Shi Wenchang Sun Yufang

(中国科学院软件研究所、中科红旗软件技术有限公司 北京 100080)
(Institute of Software, Chinese Academy of Sciences, Beijing, 100080, China;
E-mail: {rockee, yfsun}@sonata.iscas.ac.cn)

摘要 以一个称为 RS-Linux的安全操作系统的研制工作为实验手段,对按照 CC标准的思想开发安全产品的方法进行研究,并以该项研究为基础,探讨从 CC框架蕴涵的方法学中可以通过什么方式确定安全产品的安全可信度。

关键词 安全可信度 安全工程 共同标准 安全操作系统 安全系统开发
中图法分类号 TP 309; TP 316

Abstract On the basis of a research work conducted to develop a secure operating system named RS-Linux in line with the CC concepts. The investigation is conducted on what way that confidence in security may be determined in the methodologies implied in the CC framework.

Key words confidence in security, security engineering, common criteria, secure operating systems, development of secure systems

随着因特网的影响和网络应用范围的迅猛扩大,公众追求计算机安全的意识日益增强。然而,由于缺乏相应的必备知识、专业技能和支持手段,计算机系统的用户通常很难恰当地把握一个计算机安全系统的安全可信度。安全评价是使用户相信一个产品的安全性的一种方法,但我们必须注意,由评价机构发放的安全产品的评价证明书既没有增强产品的安全性,也不能说明一个产品一定能有效地解决用户所面临的安全问题[1]。我们认为,系统的开发人员和用户都有必要清楚地认识产品的安全评价所能够给我们带来的安全保证内容。

最新确立的信息安全评价国际标准(简称 CC标准)[2~4]对产品的安全评价具有良好的前景,它提倡通过安全工程的思想去确保产品的安全性。操作系统的安全职能在计算机安全中起着至关重要的作用。迄今为止,我们尚未看到讨论按照 CC标准框架研制安全操作系统的相关文献。我们以一个安全操作系统(即:红旗安全 Linux,简称 RS-Linux)的研制工作为实验手段,对按照 CC标准的思想开发安全产品的方法进行了研究,本文以该项研究为基础,探讨从 CC框架蕴涵的方法学中可以通过什么方式确定安全产品的安全可信度。

1 RS-Linux开发概述

支撑着 CC标准对计算机安全性的评价方法的一个基本思想认为,计算机安全性的可信度可以通过人们在计算机安全系统的开发、评价和使用过程中的行为体现出来。CC标准提供的是安全评价的一个框架,它没有强制指定在安全系统的开发过程中应该采用哪种具体的开发方法或生命周期模型,但它要求应该有合适的方法和模型得到应用和实施。对于一个具体的计算机安全系统的开发而言,安全工程模型可以依照瀑布模型、螺旋模型或其他软件开发模型来确定[5]。本文的讨论不针对哪一种具体的开发生命周期模型,因为这对我们的讨论没有影响,我们只关心在一个周期中的活动。

CC框架的一个显著特点是把一个计算机安全系统应该具有的安全特性与为确保这些安全特性的正确实现而采取的安全措施作为两个独立的内容进行分别对待。在研究实验中,我们主要以 LSPP[6]中确定的安全特性为基础锁定 RS-Linux的安全特性,把 RS-Linux的安全可信度定位在 EAL3增强级别。

在 CC标准框架下开发 RS-Linux所涉及到的与安全性密切相关的行为可以用图 1来描述。

2001-02-20收稿,2001-02-23修回。
* 国家 863高科技项目(863-306-ZD12-14-2)、国家自然科学基金项目(60073022)和中国科学院知识创新工程项目(KGCX1-09)资助。

CC标准用“安全评价对象”(简记为 TOE)来定义一个有待于评价的计算机安全系统及其相应的管理员和用户指南文档。这里,RS-Linux及其有关的文档就是一个 TOE,因为我们假定以后要按照 CC标准对 RS-Linux进行安全性评价。“保护轮廓定义书”(简记为 PP)和“安全对象定义书”(简记为 ST)是 CC标准中定义的两种安全需求定义结构。一个 PP针对一些确定的用户需要,为某一类 TOE定义安全需求,这些安全需求与实现无关。一个 ST为一个特定的 TOE定义安全需求和概要说明,它是相应的 TOE的评价基础。

在图 1中,我们用一个由 9个步骤构成的过程来描述 RS-Linux的开发活动,这个过程可划分成 3个阶段,即,ST定义阶段,TOE开发阶段和 TOE交货和使用阶段,其中,ST定义阶段中还包含一个 PP定义子阶段。这个过程中的每一个步骤都产生一个相应的结果,第一个结果是“安全环境”,最后一个结果是系统的“可用表示”。我们在后面还要对图 1作进一步的解释。

2 通过安全功能的确立方法确定可信度

第一个阶段(ST设计阶段)的主要目的是确立 RS-Linux的安全功能[2,7],这些安全功能锁定 RS-Linux将要提供的所有安全特性。

为达到这个阶段的目的,第一个步骤是建立系统的“安全环境”,它阐明 RS-Linux将用在什么样的环境中。安全环境定义 RS-Linux拟处理的安全问题的性质和范围,这包括安全假设、安全威胁和用户机构的安全政策等内容。就 RS-Linux而言,安全问题中要考虑的主要是安全政策,没有特别的安全威胁需要描述。

安全问题提出来之后,我们需要考虑 RS-Linux将如何对所面临的安全问题进行处理。下一个步骤是确定 RS-Linux必须满足的“安全目标”。安全目标应该是对安全问题的响应或解决方案的简明陈述。确立安全目标时,我们必须证明,对于每一个安全政策和安全假设,至少必须有一个安全目标来对它们进行处理,对于每一个安全目标,至少必须处理一个安全政策或安全假设。

第三个步骤确立能够满足已建立的安全目标的“安全需求”,特别是安全功能需求(SFR)。在可能的情况下,应尽量应用 CC标准中预定义的 SFR组件来构筑安全系统的 SFR。这里要确立的 SFR由两个方面的 SFR组成,第一个方面是能够直接满足安全目标的那些 SFR,另一个方面是被已选定的 SFR依赖的那些 SFR。例如,当 SFR1依赖 SFR2时,选定 SFR1必须同时要选定 SFR2,使得所确立的 SFR既能实现安全目标又能满足 SFR间的依赖关系。我们必须证明,对于 RS-Linux的每一个安全目标,至少必须有一个 SFR用于实现它,对于每一个 SFR,至少必须用于实现 RS-Linux的一个安全目标。我们还必须证明,所有 SFR的依赖关系是得到满足的,所有 SFR之间是相互支持的,口令机制的安全功能强度是符合安全目标要求的。

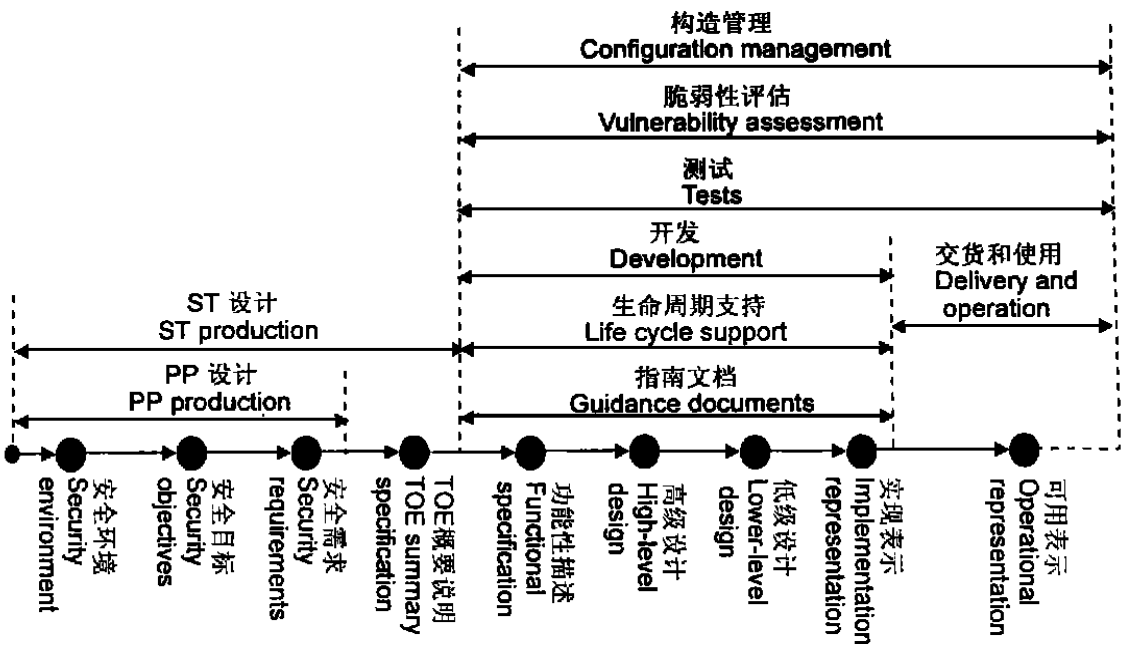

图 1 研制 RS-Linux的开发活动

Fig. 1 Development activities for building RS-Linux

这个阶段的最后一个步骤要形成 TOE概要说明，概要说明定义出满足所有 SFR要求的 RS-Linux的所有安全功能。安全功能可以刻划为 RS-Linux要满足的 SFR的精确化表述，与 SFR相比，安全功能对 RS-Linux的安全特性给予更详细的描述，使得 RS-Linux的安全特性更容易被理解。我们必须证明，对于每一个 SFR,至少必须映射到一个安全功能，对于每一个安全功能，至少必须映射到一个 SFR,并且，安全功能之间必须是相互支持的。

3 通过安全保证措施确定可信度

在第一个阶段里，我们也需要制定确保 RS-Linux的安全性的安全保证措施，但这些保证措施是在第二个阶段 (开发阶段) 和第三个阶段 (交货和使用阶段) 实施的。

在第二和第三个阶段，CC标准区分一个安全系统的 5种表现形式，它们分别是：功能性描述、高级设计、低级设计、实现表示和可用表示。

在 CC标准中，一个 TOE中实施安全性的部分简记为 TSF。功能性描述是在较高层次上对 TSF中用户看得见的接口和行为的描述。高级设计把 TSF划分为子系统进行描述。低级设计从模块的角度对 TSF的内部工作机理进行描述。RS-Linux的实现表示就是系统的源代码。RS-Linux的可用表示就是在计算机上运行的 RS-Linux操作系统，这样的系统可以运行在开发人员的工作场所，也可以运行在用户的工作场所。

在 RS-Linux的开发活动中，我们需要考虑 7类安全保证措施，即，开发类 (ADV)、生命周期支持类 (ALC)、指南文档类 (AGD)、测试类 (ATE)、构造管理类 (ACM)、脆弱性评估类 (AVA)、以及交货和使用类 (ADO)，如图 1所示。

ADV类需要开展的工作涉及非形式化的功能性描述 (ADV_FSP.1)、安全性实施的高级设计 (ADV_HLD.2)、非形式化的一致性证明 (ADV_RCR.1)、以及非形式化的安全政策模型化 (ADV_SPM.1)等方面的安全保证措施。ADV_FSP.1措施为系统提供非形式化的功能性描述。ADV_HLD.2措施提供系统的高级设计，把系统的安全性实施子系统与系统的其它部分区别开来。对于要求提供的 TSF的所有各种表示形式，ADV_RCR.1措施分析所有相邻的两种表示之间的一致性。具体地说就是，安全功能与功能性描述之间、功能性描述与高级设计之间、高级设计与源代码之间等的一致性；这些分析必须证明，相对抽象的一种表示中蕴涵的所有相关安全特性，必须在相对具体的对应表示中全部正确地得到细化，比如，功能性描述中蕴涵的所有安全特性，必须在高级设计中全部正确地得到细化。安全保证等级 EAL3不要求提供低级设计。ADV_SPM.1措施为所有可用模型表示的安全政策提供非形式化的安全政策模型，并证明系统的功能性描述与安全模型间的一致性。

ALC类需要开展的工作涉及安全措施的陈述 (ALC_DVS.1) 等方面的安全保证措施。ALC_DVS.1措施描述并实施在开发环境中应采取的安全措施，开发环境中的安全措施保护 RS-Linux的设计与实现的完整性，其目的是消除或降低开发环境中可能存在的对安全性的威胁。

AGD类需要开展的工作涉及管理员指南 (AGD_ADM.1) 和用户指南 (AGD_USR.1) 等方面的安全保证措施。AGD_ADM.1措施为 RS-Linux的安全性管理提供管理员指南。AGD_USR.1措施提供用户指南，用户指南描述用户可用的 RS-Linux的安全功能的正确使用方法。

ATE类需要开展的工作涉及覆盖范围分析 (ATE_COV.2)、高级设计测试 (ATE_DPT.1)、功能性测试 (ATE_FUN.1)、以及独立抽样测试 (ATE_IND.2) 等方面的安全保证措施。ATE_COV.2措施证明所指定的测试能覆盖功能性描述中给出的所有安全功能。ATE_DPT.1措施确保所指定的测试能证明 RS-Linux的工作行为与高级设计中确定的情况一致。ATE_FUN.1措施确保所指定的测试能证明所有的安全功能都按指定的方式进行工作。ATE_IND.2措施确保第三方能对开发人员所做的测试进行抽样重复测试，并得到开发人员所描述的测试结果。

ACM类需要开展的工作涉及授权控制 (ACM_CAP.3) 和 TOE构造管理覆盖范围 (ACM_SCP.1)等方面的安全保证措施。ACM_CAP.3措施确保 RS-Linux的所有构造项目都得到妥善的管理，对构造项目的任何修改都是在经过授权的情况下进行的。ACM_SCP.1措施确保最低限度也要对以下构造项目进行追踪管理：RS-Linux源代码、设计文档、测试文档、用户文档、管理员文档、以及构造管理文档。

AVA类需要开展的工作涉及指南审查 (AVA_MSU.1)、TOE安全性强度评价 (AVA_SOF.1)、以及开发人员对脆弱性的分析 (AVA_VLA.1)等方面的安全保证措施。AVA_MSU.1措施确保文档中不存在误导性的、不合理的或冲突性的指南。AVA_SOF.1措施确保由口令机制实现的安全功能能够达

到指定的强度级别。AVA_VLA.1措施确保开发人员对 RS-Linux的所有可发行媒介都进行检查，排查用户有可能违反安全政策的任何明显途径，并证明没有任何明显的脆弱性会在 RS-Linux的使用环境中被利用。

ADO类需要开展的工作涉及交货程序（ADO_DEL.1）、以及安装、生成和启动程序（ADO_IGS.1）等方面的安全保证措施。ADO_DEL.1措施提供发行 RS-Linux的交货程序，以便按照该交货程序能把 RS-Linux安全地送到用户手中。ADO_IGS.1措施提供安全地安装、生成和启动 RS-Linux的操作程序。

值得注意的是，CC标准只预定义了可供使用的安全保证需求组件和可供选定的安全保证需求，但它没有也不可能规定具体的安全保证措施。制定安全保证措施是系统开发人员的职责。

4 讨论

以图 1中描述的三阶段开发活动模式为讨论的基础，本文的第 2节指出，第一个阶段首先对安全问题进行描述，然后试图通过逐步渐进的方式定义一组能够有效地解决相应安全问题的安全功能。CC标准对 ST的评价将证实第一个阶段定义的安全功能能有效地解决相应的安全问题。因而，如果一个 ST通过了 CC标准的评价[8]，那么，这个 ST中定义的安全功能能够处理该 ST中描述的安全问题，这一点是可信的。

本文的第 3节指出，为了在一定程度上确保一个安全系统能够实现指定的安全功能，安全保证措施在第二和第三个阶段付诸实施。CC标准对一个安全系统的评价是以一个已通过评价的 ST为参考依据的。一个安全系统通过了 CC标准评价的证明可以证实相应的所有安全保证措施都在第二和第三个阶段有效地得到了实施。由此可以断定，如果一个安全系统通过了 CC标准的评价，那么，这个安全系统就在一定程度上正确地实现了相应 ST中定义的安全功能，进而，这个安全系统就能够在一定程度上解决相应 ST中描述的安全问题。

这里，我们说，一个安全系统能够在一定程度上解决某些安全问题，这个程度就是安全可信度的重要衡量指标，它是由在开发活动中实施的安全保证措施决定的。我们必须对安全保证措施加以了解，才能认识一个安全系统的安全性在哪些方面以及在多大程度上是值得相信的。

5 结语

通过把合适的安全工程方法应用到安全功能的确立和安全保证措施的实施活动中，我们可以确立一个安全系统的安全可信度。一般而言，通过 CC标准评价的安全系统具有相对高的安全可信度。然而，从 CC框架下的安全系统开发过程可见，每个安全系统都是以确定的安全问题为处理对象的。离开一个系统所要针对的安全问题的范围，安全系统就没有安全性可言，不管该安全系统是否通过安全性评价。

参考文献

1 Steve Lipner. Twenty years of evaluation criteria and commercial technology. Proceedings of the 1999 IEEE Symposium on Security and Privacy, Oakland California, 1999. 9 ~ 12.

2 The International Organization for Standardization and the International Electrotechnical Commission, Joint Technical Committee 1. Evaluation Criteria for IT Security-Part 1: Introduction and General Model, 1999 (E): ISO/IEC 15408-1.

3 The International Organization for Standardization and the International Electrotechnical Commission, Joint Technical Committee 1. Evaluation Criteria for IT Security-Part 2 Security Functional Requirements. 1999 (E): ISO/IEC 15408-2.

4 The International Organization for Standardization and the International Electrotechnical Commission, Joint Technical Committee 1. Evaluation Criteria for IT Security-Part 3 Security Assurance Requirements. 1999 (E): ISO/IEC 15408-3.

5 Marshall D. Abrams, Security engineering in an evolutionary acquisition environment, Proceedings of the 1998 Workshop on New Security Paradigms, 1998. 22~ 26, Charlottesville, VA USA.

6 Information Systems Security Organization. Labeled Security Protection Profile, National Security Agency, Fort George G Meade MD, USA, 1999.

7 The International Organization for Standardization and the International Electrotechnical Commission, Joint Technical Committee 1. Guide for Production of PPs and STs, 1998, Version 0.6, ISO/IEC JTC 1/SC 27/WG 3 N452.

8 Common Criteria Interpretation Management Board. Common evaluation methodology for information technology Security, Part 2 evaluation methodology. Common Criteria Project Sponsoring Organizations, 1999, Version 1.0, CEM-99/045.

(责任编辑：黎贞崇)