OpenChain安全保证规范1.1

（中文版。翻译人：张俊霞，中国信息通信研究院）

OpenChain项目：自2016年以来，致力于建立可信供应链

规范介绍

OpenChain项目正在努力建立一个供应链，确保开源软件在传递中具有可信、一致的合规信息。作为这项任务的一部分，我们维护《OpenChain ISO/IEC 5230:2020》规范，这是开源许可协议合规的国际标准。下一步，自然是以一个更广阔的视野，制定一份定义和展示在使用开源软件时每个安全保证程序应满足的最低核心要求的指南。

从内容上看，《OpenChain ISO/IEC 5230:2020》规范是一个流程管理规范，用于确定应该存在流程、策略或培训的入站、交互和出站拐点。同时，识别和跟踪我们使用和部署的软件是实现此目标不可或缺的部分，这使我们的方法对安全或出口控制也很有用。

OpenChain项目社区注意到《OpenChain ISO/IEC 5230:2020》正在安全领域得到应用，并决定开发此安全规范以满足市场需求。本规范（《OpenChain安全保证规范》）旨在定义和描述使用开源软件背景下质量安全保证程序的核心要求。它专注于一个狭窄子集：根据CVE、GitHub/GitLab漏洞报告等众所周知的安全漏洞来检查开源软件。

《OpenChain安全保证规范》侧重于质量安全保证程序中的“什么”和“为什么”方面，而不是深入研究“如何”和“何时”。这是一个有意识的决定，旨在确保任何规模和任何市场的组织都能灵活地使用此参考规范。这种方法以及确定的流程类型基于全球众多公司围绕此类程序的创建和管理五年多的实用的反馈。这也使得公司可以制定一个完全符合其自身供应链要求的程序，范围可以为单个产品或完整的法律实体，并快速有效地将该解决方案推向市场。

本规范基于2022年3月28日发布的安全保证参考指南2.0（RC 1）。该参考指南通过我们的常规投票实践进入了最终批准过程，成为本正式发布和出版的安全规范。本参考规范的范围可能在后续根据社区反馈而扩大。

本规范的导言描述了其目的。第2节定义了本文档中使用的关键术语。第3节定义了为实现安全保证的核心水平的安全保障程序必须满足的要求。每个要求都由一个或多个验证材料（即记录）组成，这些材料必须可展示，以证明满足了要求。验证材料不需要公开，尽管组织可能会选择将其提供给其他人，例如也可根据保密协议（NDA）来提供。

OpenChain安全保证规范

1：范围

本规范规定了合格的开源软件安全保证程序的关键要求，该程序在组织间交换由开源软件组成的软件解决方案时可建立信任。

2：术语、定义和示例

就本文件而言，适用以下术语和定义。

2.1 - 组件记录

组件记录应包括供应商名称、组件名称、组件版本、其他唯一标识符、依赖关系、SBOM数据作者和时间戳（根据NTIA《SBoM最小元素》）

2.2 - 客户协议

供应链上下游一致认为，相关客户或用户组织（如果适用）认为用于查找、跟踪或修复安全问题的流程是充分和正确的。

2.3 - CVE

Common Vulnerabilities and Exposures（CVE）是一个公开的计算机软件安全问题和缺陷的公共数据库。当有人提到CVE时，他们指的是在数据库中分配了CVE ID号码的安全漏洞。CVE数据库由美国国土安全部（DHS）和网络安全和基础设施安全局（CISA）赞助。

2.4 - 记录在案的证据

这是明确存储的信息，概述、解释或记录了与本规范中提到的活动和行动相关的信息。

2.5 - 已知漏洞

以往的、在公开可用的开源软件组件中发现的安全漏洞。这将包括任何公开发布的漏洞，包括但不限于CVE、GitHub/GitLab漏洞警报、软件包管理器警报等。

2.6 - 新发现的漏洞

刚刚在公开可用的开源软件组件中发现的安全漏洞。这将包括任何公开发布的漏洞，包括但不限于CVE、GitHub/GitLab漏洞警报、软件包管理器警报等。

2.7 - 开源软件

受一个或多个许可协议约束的软件，这些许可协议符合开源软件促进会OSI发布的开源软件定义（见www.opensource.org/osd）或自由软件基金会发布的自由软件定义（见www.gnu.org/philosophy/free-sw.html）。

2.8 - 程序

构成组织安全保证活动的一套政策、流程和人员。

2.9 - 程序参与者

定义、贡献或负责准备提供的软件的任何组织员工或承包商。注意：根据组织的不同，这可能包括（但不限于）软件开发人员、发布工程师、质量工程师、产品营销、产品管理和采购。

2.10 - 安全保证

对系统满足安全最佳实践要求并能够抵御已知漏洞的可信度。

2.11 - 安全测试

软件（或其他组件）分析的过程，通过该过程可了解在已知漏洞的背景下，软件当前和潜在未来的管理。

2.12 - 软件物料清单（SBOM）

SPDX（ISO/IEC
5962:2021）等结构化格式的信息，允许为软件包交换信息，该软件包应以对第三方有用的方式，能有效地包含名称、版本、来源、许可协议、版权和已知漏洞。

2.13 - 提供的软件

组织分发或提供给第三方（例如其他组织或个人）的软件。

2.14 - 验证材料

证明满足规范特定要求的材料。

3 - 要求

3.1 - 程序基础

3.1.1 - 政策

创建一个书面政策，用于管理所提供软件的开源软件安全保证。该政策将在内部传达。该政策及其沟通方法将有一个审查过程，以确保它们是现行的和相关的。

验证材料：

- 3.1.1.1：记录在案的开源软件安全保证政策；
- 3.1.1.2：记录在案的使计划参与者了解安全保证政策的过程。

理由：

这是为了确保存在一个创建、记录和让计划参与者意识到开源软件安全保证政策的流程。虽然这里没有就政策中应该包含的内容提供任何要求，但其他章节可能会提出附加的要求。

3.1.2 - 能力

该组织应：

- 定义影响该程序绩效和有效性的角色和责任；
- 确定项目参与者履行每个角色的必要能力；
- 确保项目参与者拥有适当的教育、培训和/或经验；
- 在适用的情况下，确保项目参与者采取行动获得必要的能力；
- 保留适当的书面信息，作为能力以及目前谁是该计划参与者的证据。

验证材料：

- 3.1.2.1：一份书面列表，记录不同项目参与者的角色和应具有的相应职责；
- 3.1.2.2：一份书面文件，定义每个角色的能力；
- 3.1.2.3：参与者名单及其角色；
- 3.1.2.4：每个项目参与者进行评估能力的书面证据；
- 3.1.2.5：定期审查和更改流程的书面证据；
- 3.1.2.6：记录在案的验证文件，确认现有安全保障计划是否与公司内部最佳实践有关，以及由谁负责完成这些流程。

理由：

确保项目参与者有足够的能力来履行各自的角色和责任。

3.1.3 - 意识

组织将确保项目参与者了解：

- 开源软件安全保证政策；
- 相关程序目标；
- 他们对该程序有效性的贡献；

- 不遵守该程序要求的影响。

验证材料：

- 3.1.3.1：评估项目参与者意识的书面证据——
  其中应包括计划的目标、一个人在计划中的贡献以及未按程序执行的影响。

理由：

为了确保项目参与者对各自在计划中的作用和责任有足够的认识。

3.1.4 - 计划范围

程序应定义与整个组织的风险管理政策相匹配的指导原则和范围。应该清楚该计划适用于产品线、部门还是整个组织。还应该理解，这个范围可能会随着时间的推移而变化，指标可用于评估其持续有效性。

验证材料：

- 3.1.4.1：明确定义该程序范围和限制的书面声明；
- 3.1.4.2：该程序应实现的一套改进指标；
- 3.1.4.3：每次审查、更新或审计的书面证据，以证明程序的持续改进。

理由：

提供灵活性，以构建最适合组织需求范围的程序。一些组织可以选择为特定产品线维护程序，而另一些组织可以选择实施程序来管理整个组织的提供的软件。

3.1.5 - 标准实践实施

该计划通过定义和实施以下程序，演示了对已知漏洞和安全软件开发的健全和强大的处理程序：

- 定义了识别所提供软件的结构和技术威胁的方法；
- 检测供应软件中存在已知漏洞的方法；
- 跟进已确定的已知漏洞的方法；
- 在必要时向客户群传达已识别的已知漏洞的方法；
- 分析供应软件发布后新发布的已知漏洞的供应软件的方法；
- 在发布前，对所有提供的软件都应用了连续和重复的安全测试方法；

- 验证已识别风险在发布提供软件之前是否已解决的方法；
- 酌情向第三方输出有关已识别风险的信息的方法。

上述安全保证方法应存在明确的流程。

验证材料：

- 3.1.5.1：上述每种方法都有一个记录在案的流程。

理由：

确保存在适当的流程来检测和跟进提供的软件中的已知漏洞。

3.2 - 相关任务的定义和支持

3.2.1 - 访问

维护一个流程，以有效响应已知漏洞外部查询。定义一个公开的第三方可以查询特定软件产品已知漏洞的方法。

验证材料：

- 3.2.1.1：允许第三方查询已知漏洞或新发现漏洞的公开途径（例如，通过由计划参与者监控的电子邮件地址或门户网站）；
- 3.2.1.2：存在内部记录程序，用于回复第三方已知漏洞或新发现漏洞的查询。

理由：

确保第三方有合理的方式就安全漏洞查询可以与组织安全的进行联系，并确保组织准备做出回应。

3.2.2 - 有效的资源

定义和资源计划任务：

- 分配责任，以确保成功执行程序任务；
- 程序任务享有足够的资源；
- 已经分配了足够的时间来执行任务；
- 已经分配了充足的资金；

- 具有审查和更新政策和支持任务的过程；
- 任何可能需要此类指导的人可以获得与已知漏洞相关的技术专业知识。

验证材料：

- 3.2.2.1：注明项目角色中人员、团体或职能的文件；
- 3.2.2.2：已确定的方案角色配备了适当的人员，并提供了充足的资金；
- 3.2.2.3：确定可用于解决已确定的已知漏洞的专业知识；
- 3.2.2.4：分配安全保证内部责任的书面程序。

理由：

确保：i）计划责任得到有效支持和资源，ii）定期更新政策和支持流程，以适应安全保证最佳实践的变化。

3.3 - 开源软件内容审核和批准

3.3.1 - 软件物料清单（SBOM）

应存在创建和维护SBOM的过程，其中包括所提供软件的每个开源软件组件。

验证材料：

- 3.3.1.1：一个记录在案的过程，确保在所提供的软件整个生命周期内不断记录所提供软件中使用的所有开源软件。这包括所提供软件中使用的所有开源软件的存档；
- 3.3.1.2：所提供软件的开源软件组件记录，证明正确遵循了记录的程序。

理由：

确保存在创建和管理用于构建所提供软件的软件物料清单的过程。需要一份SBOM来支持对每个组成部分的系统审查，以了解是否存在任何已知漏洞。

3.3.2 - 安全保证

- 对于所提供软件版本SBOM中的每个开源软件组件进行审查；
- 应用检测已知漏洞存在的方法；
- 对于每个已确定的已知漏洞，分配一个风险/影响评分；

- 对于每个检测和分配的分数，确定并记录适用于软件用例的必要补救步骤，并在先前确定的级别或更高级别时获得客户协议（例如，所有严重性分数高于4.5...）；
- 根据风险/影响评分，采取适当的行动（例如，必要时联系客户，升级软件组件，无需进一步行动，...）；
- 如果之前分发的软件中存在新发现的漏洞，根据风险/影响评分，采取适当的行动（例如，如有必要，请联系客户）；
- 能够在提供的软件发布上市后对其进行监控，并响应已知漏洞或新发现的漏洞披露。

验证材料：

- 3.3.2.1：处理检测和解决所提供软件开源软件组件已知漏洞的文档化过程；
- 3.3.2.2：对于每个开源软件组件，都会保存已识别的已知漏洞和所采取的行动的记录（即使不需要操作）。

理由：

确保程序具有足够的鲁棒性，能够处理所提供软件所属开源软件的已知漏洞。存在支持此活动的过程，并遵循该过程。

3.4 - 遵守准则要求

3.4.1 - 完整性

要使程序被认为符合本规范，组织应确认该计划符合本文件中提出的要求。

验证材料：

- 3.4.1.1：确认§3.1.4中规定的程序的书面证据符合本文件的所有要求。

理由：

确保如果一个组织声明其有一个符合要求的程序，则该程序符合本文件的所有要求。仅仅满足这些要求的子集被认为是不够的。

3.4.2 - 持续时间

符合此版本规范的程序将有以下审查期：从第一次认证开始18个月，从第二次认证开始24个月，从第三次认证开始36个月。此后，它需要每36个月审查一次。

验证材料：

- 3.4.2.1：一份确认该计划在获得一致性验证后的18个月内符合本规范所有要求的文件。

理由：

如果一个组织希望随着时间的推移保持一致性，程序必须随时了解规范要求。如果组织随着时间的推移继续维护计划的一致性，这一要求确保了该计划的支持流程和控制不会受到侵蚀。

如果一个组织希望在一段时间内保持一致性，那么应保持与规范要求一致且正在执行是至关重要的。这一要求确保了如果一个组织在一段时间内继续宣称项目符合性，则项目的支持过程和控制不会受到侵蚀。