基于SBOM的软件安全治理实践

Practice of Software Security Governance Based on SBOM

王　戈[1,2]，郭新海[1,2]，刘　安[1,2]，丁　攀[1,2]，蓝鑫冲[1,2]（1. 中国联通研究院，北京 100048；2. 下一代互联网宽带业务应用国家工程研究中心，北京 100048）

Wang Ge[1,2], Guo Xinhai[1,2], Liu An[1,2], Ding Pan[1,2], Lan Xinchong[1,2] (1. China Unicom Research Institute, Beijing 100048, China; 2. Next Generation Internet Broadband Service Application National Engineering Research Center, Beijing 100048, China)

摘　要：

当今高度信息化和数字化时代，软件已经成为人们生活和工作中不可或缺的重要组成部分。随着软件产业的快速发展和安全事件频繁发生，软件自身的安全问题已经成为当下亟待解决的重大问题。针对这一挑战，越来越多的企业和组织开始关注软件安全治理，其中基于SBOM的软件安全治理方案效果显著，得到了越来越多的关注和实践。

关键词：

SBOM；软件供应链；安全治理；开源组件

doi：10.12045/j.issn.1007-3043.2023.08.003

文章编号：1007-3043（2023）08-0009-05

中图分类号：TN915.08

文献标识码：A

开放科学（资源服务）标识码（OSID）：

Abstract：

In today's highly informationized and digitized age, software has become an essential and integral part of people's daily life and work. Due to the fast development of the software industry, and frequent occurrence of security incidents, software security issues have become a major problem that urgently needs to be solved. To address this challenge, more and more enterprises and organizations are focusing on software security governance. Among them, the software security governance solution based on SBOM has shown significant effectiveness and gained more attention and practice.

Keywords：

SBOM; Software supply chain; Security governance; Open-source component

引用格式：王戈，郭新海，刘安，等. 基于SBOM的软件安全治理实践[J]. 邮电设计技术，2023(8)：9-13.

1 概述

随着软件在社会经济生活中越来越广泛地应用，软件供应链安全问题日益引起人们的重视。为了实现软件的快速开发和应用，当前软件绝大部分都是采用组件组装而成，且随着开源文化的兴起，开源组件在软件中的使用比例迅速升高。开源组件的引入虽然加快了软件开发与迭代效率，同时也将开源的安全问题引入了软件供应链[1]。针对当前开源软件的大量使用，攻击者会通过网络工具、下载投毒、代码污染、漏洞利用等供应链攻击手段对企业的软件系统进行破坏性攻击[2]。近几年此类攻击事件频发，攻击手段多样，其中影响巨大的安全事件有SolarWinds攻击、Realtek Wi-Fi SDK漏洞、Apache Log4j2漏洞等，这些事件都给企业和用户带来了重大的损失。

为了解决上述软件供应链的安全问题，涌现出许多新的软件安全治理方法，其中一种基于软件物料清单（Software Bill of Materials，SBOM）的软件治理方法得

收稿日期：2023-07-03

到越来越多的重视[3]。SBOM明确描述了一个软件产品中包含的所有组件及其依赖关系，帮助提高软件的透明度和信任度，可以很好地帮助治理软件供应链所面临的安全问题，构建一个更加安全的软件供应链体系。

2 软件安全治理现状

当前，软件供应链的攻击事件日益增多，软件正面临着严重的安全威胁，如何有效地降低软件供应链的安全风险，已经成为当前国内外机构、企业和组织重点关注的问题[4]。我国相关部门、机构和组织已通过在软件供应链安全领域积极布局政策法规、标准体系以及技术能力的研究和建设来应对日益频繁和复杂的软件供应链安全风险[5]。当前，软件安全治理仍然存在如下突出问题。

a）软件成分信息不透明。如今软件结构越来越复杂，开源组件的使用占比越来越高，在重点行业的使用率已接近90%[6]，使得软件成分难以梳理，组件间的关系混乱，导致软件的安全治理无从下手。

b）漏洞识别不全且修复成本高。由于软件的成分复杂，漏洞多样，有效地确定软件漏洞仍然存在挑战，另外，针对升级组件来修复的漏洞，在不明确软件内组件依赖关系的情况下对组件进行升级，可能会因为兼容性问题导致升级失败和漏洞无法修复。

c）知识产权不合规。使用开源软件仍然需要遵循相关的开源协议，目前软件开发中忽略许可证信息的现象非常普遍，这样会引发知识产权的相关问题[7]。

综上所述，软件安全治理当前面临的核心问题还是软件成分的不透明。下面将针对上述问题的特点，引入SBOM概念，就如何治理软件风险展开论述。

3 SBOM介绍

3.1 SBOM的概念与定义

根据美国国家电信和信息化管理局（NTIA）的定义，SBOM是一份包含软件的所有组件的信息和层级关系的形式化、机器可读清单[8]。SBOM其实就是一个结构化列表，用来描述组成某个特定软件或组件的所有成分及其关系。近几年，SBOM被广泛地应用到了软件安全领域，以支持制定安全策略、风险评估、漏洞管理等工作。

3.2 SBOM的元素

SBOM是软件组成成分的一个列表，其中包含了软件基本信息、软件间的关系和软件其他信息三大类软件成分的信息。在实践过程中，用户可以根据实际情况在表1的软件基本信息的基础上增加配置扩展信息字段，形成适用于自身的SBOM清单。

3.3 SBOM的格式

国际上主流的SBOM格式有以下3种：软件包数据交换（Software Package Data Exchange，SPDX）、软件识别标记（Software identification，SWID）和依赖关系交换（Cyclone Dependency eXchange，CycloneDX）。

SPDX是一个ISO/IEC标准格式，用于交换软件物料清单信息[9-10]。SPDX的特点是对许可证的详细信息支持较好，主要支持的输出文件格式如下：RDF、XLS、SPDX、YAML、JSON。

CycloneDX是一个轻量级SBOM规范，可用于应用

表1　SBOM元素信息表

类型		项目	说明
软件基本信息		作者信息	创建组件SBOM数据的实体信息
		时间戳	SBOM最后一次更新的日期和时间
		供应商名称	创建、定义和识别组建的实体名称，也可以为其标识符
		组件名称	由原始供应商定义的软件单元名称
		组件版本	供应商用于标识软件版本变化的信息
		组件哈希值	用于标识组件文件的唯一性
		唯一标识符	CPE、URL（PURL）、UUID、SWHID和组件哈希值
软件扩展信息	软件间的关系	依赖关系	用于描述软件包含上游组件的关系，例如：includes
		包含关系	如源代码与编译后二进制的包含关系，发布容器镜像与二进制的包含关系等
		其他关系	其他关联关系
	软件其他信息	软件知识产权信息	包括开源许可证版权与开放标准、第三方授权信息等
		关联漏洞信息	漏洞信息，如对应CVE、CNVD、CNNVD等
		备注	-

程序安全上下文和供应链组件分析，主要输出格式包括XML、JSON。

SWID是一个标准化的XML格式，用来标识产品、版本、产品生产分发中的组织和个人、组件信息、产品和其他描述性元数据之间的关系等信息。

上面描述的3种SBOM格式各具特点，在实际的使用过程中，需要用户根据自身情况，选取合适的格式，作为统一格式对所有维护软件的SBOM进行管理。

4 基于SBOM的软件安全治理

SBOM作为一种技术手段在软件行业中逐渐被接受和推广，其主要价值在于提高软件生态透明度和安全性。因此，本文将SBOM引入到软件安全治理中，与已有的安全能力和工具配合使用，降低软件的风险。图1所示为基于SBOM的功能和特点衍生出的软件安全治理的相关能力及具体举措。下面本文着重介绍如何在软件治理过程中运用SBOM及可以起到的作用。

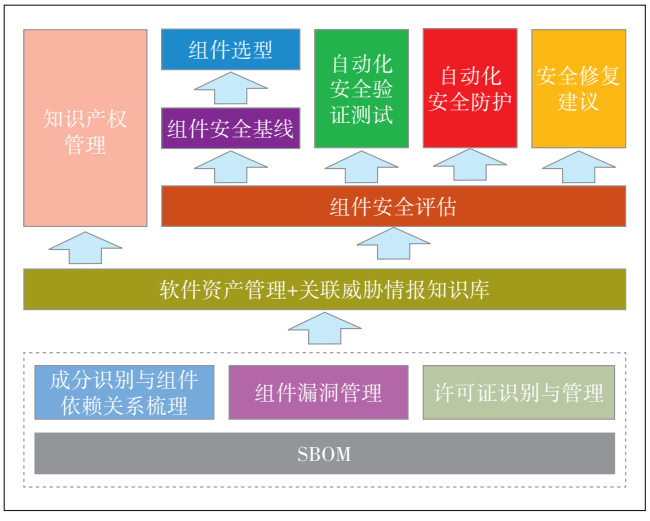

图1　基于SBOM的软件安全治理能力架构

4.1 SBOM在软件安全治理中的作用

SBOM在软件安全治理中有如下功效：梳理组件之间的依赖关系、监控组件的漏洞、管理组件许可证。

a）梳理组件之间的依赖关系。SBOM不仅可以生成软件的成分清单，还可以提供清晰的组件依赖关系图，帮助挖掘出原先难以发现的间接依赖组件，便于规范组件的评估和使用，在软件的运维过程中，也更有助于对软件的管理和维护，为各项运维操作提供评估参考和依据，降低在软件变更或升级时的安全风险。

b）监控组件的漏洞。由于SBOM具有清晰的软件物料清单和梳理组件依赖关系的特点，SBOM在软件漏洞的管理中可以起到非常重要的作用，在软件结构复杂、组件引用关系混乱的情况下进行软件安全漏洞的定位、整改和修复，提升软件的内生安全性。后面章节会具体详述如何利用SBOM对软件漏洞进行管理。

c）管理组件许可证。使用开源软件同样需要遵循软件的开源协议，违反开源许可证也已经被越来越多的国家法律认定为侵权行为。为了降低软件侵权风险，企业需要识别软件中涉及的许可证，有效管理项目中涉及的许可证风险。SBOM可以明确梳理出组件所遵循的许可证协议，帮助管理软件的知识产权风险。

4.2 SBOM在软件安全治理中的应用

SBOM在软件全生命周期的各个阶段都能发挥重要的作用，帮助提升软件的质量，下面将会从软件资产管理、软件安全评估、软件安全运营及防护和软件授权管理4个维度详细介绍SBOM是如何在软件安全治理中运用并发挥作用的。

4.2.1 软件资产管理

软件资产的梳理是软件治理的基础。SBOM本身的作用就是提供软件构成的物料清单，软件使用了哪些组件、组件相关的信息以及组件间的依赖关系都可以清晰地提供，正好可以协助完成软件资产的收集和管理工作。图2所示为软件资产管理的总体架构。

a）建立完整的组件资产清单台账。根据软件的SBOM清单，梳理软件中引用的开源组件、第三方商业组件、自有组件以及这些组件间接引用的所有组件信息，并加入软件自身的供应商相关信息、软件类型、API接口信息、开发语言、开发工具、中间件、数据库以及开发框架等软件相关信息共同构建软件资产清单并形成台账，作为后续对软件安全治理的基础和依据。

b）构建软件资产更新机制。设置清单定期自动更新、软件代码变更或二进制文件发生变化后自动同步更新资产清单，以及软件上游供应商的SBOM更新后同步更新资产清单。保证软件资产清单与最新情况相吻合，确保后续基于软件资产清单的工作不会出现偏差。

c）将软件资产清单对接威胁情报知识库。软件资产管理的最终目的还是要对软件的安全性做评估

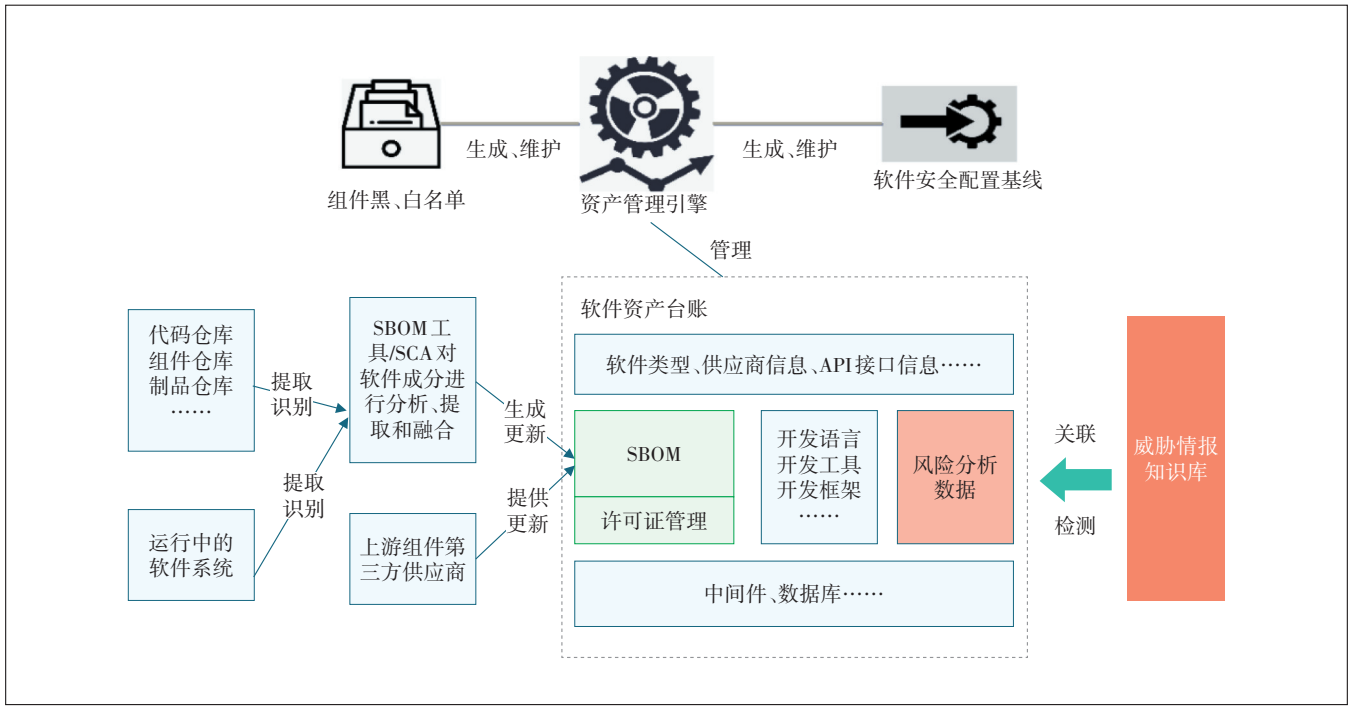

图2　软件资产管理架构

和管理。将软件资产清单关联威胁情报知识库后，可以帮助对清单中的成分进行安全性评估，及时检测软件成分是否存在安全风险及漏洞，提供修复建议和整改方案等，降低软件安全风险。

d）依据构建的软件资产清单建立并维护基线。根据已有的软件资产清单对组件进行安全评估，设置安全基线，并根据安全基线配置组件的白名单和黑名单；同时根据资产清单中软件框架、配置等相关信息，综合评估并设置软件的安全配置基线。此外，根据软件资产的自更新机制，定期对组件的安全基线和黑、白名单以及软件的安全配置基线等进行定期维护，保证新增风险也能纳入监管之中。通过建立和维护基线，规范化组件的的使用和配置，为组件的引用和配置提供合规依据。

e）知识产权管理。基于SBOM的组件许可证自动识别、兼容性检查及管理能力，构建软件的知识产权管理能力，实现对软件中直接引用和间接引用的组件许可证及权限的综合管理以及软件自身的许可证生成与维护，确保软件的权限合法合规，避免软件面临知识产权风险。

4.2.2 软件安全评估

基于SBOM的特性，在软件生命周期的各阶段都可以帮助对软件的安全性进行评估。下面就比较典型的需求设计阶段的组件选型和安全测试阶段的组件安全测试2个场景介绍SBOM在软件安全评估中的应用。

a）需求设计阶段的安全评估。可以根据具体的功能需求选取几款可以满足要求的组件作为候选，然后结合软件的资产管理，依据备选组件的SBOM对组件的安全性进行分析和评估，最后再根据设置的安全基线和组件黑白名单挑选出符合要求的组件作为最终所使用的组件，保证所选组件合规、安全。

b）软件测试阶段的安全评估。如图3所示，管理平台基于SBOM关联威胁情报知识库生成被测试软件的安全风险报告，并依据此报告中的风险点及详细信息生成验证测试用例，之后可以采用人工渗透测试，或者使用自动化的测试工具，如IAST等，按照测试用例对软件进行测试并生成最终的安全测试报告，这样有针对性的风险验证测试可以大大提高测试效率和精度，非常适合如今敏捷开发、快速迭代的开发模式。

4.2.3 软件安全运营与防护

在软件的上线运营阶段，建立软件的安全持续监控和防护体系，利用软件资产清单，关联威胁情报知识库，对清单上的软件成分进行自动化安全监控，当发现清单上的组件存在安全风险时，快速定位问题组件，找到受影响的软件。同时，管理平台从威胁情报

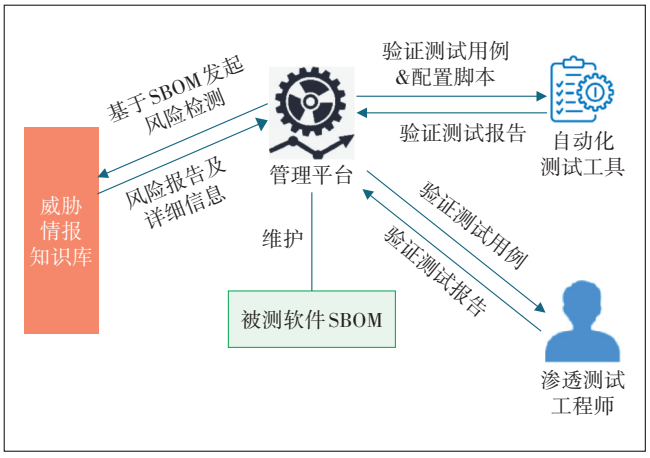

图3　测试阶段软件人工/自动化安全风险评估

知识库获取风险的原理及利用方式等详细信息，再结合软件的资产清单中配置与运行环境等信息，给出综合的修复和防护方案。如果此时已对接安全防护工具[11]（例如RASP或WAF等），管理平台则可以在第一时间将防护方案或配置脚本推送防护工具并执行，实现自动化安全防护，减少安全风险的暴露时间，降低软件的风险。之后可以再根据给出的修复方案对问题组件进行升级或修复，确保问题彻底解决。图4所示为基于软件资产清单的软件风险检测与自动化防护和修复过程示意。

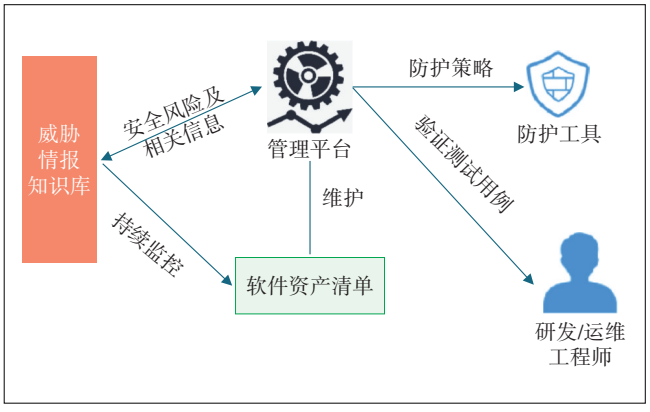

图4　软件风险检测与自动化防护和修复

5　结束语

本文着重介绍了基于SBOM使软件成分透明化的特性以及SBOM在软件安全治理中发挥的作用及相应的应用实践。将SBOM引入软件安全治理，彻底解决了软件成分信息难以获取的问题，打开了组件安全治理的一扇大门，未来基于SBOM的软件安全治理方案、治理工具、治理能力也会随着技术的更新和进步得到快速发展和应用。然而，SBOM的使用中也存在自身的安全风险问题，如何保证SBOM在存储和传输的过程中不被恶意篡改，如何保证SBOM只能具有权限的人才能访问，防止被黑客获取后进行针对性的攻击等等。随着今后SBOM使用的普及，更多的安全防护技术、数据安全技术以及密码技术等也会与SBOM技术结合使用，保障SBOM在生成、传输、使用和存储过程中的安全性和可靠性，不断提升软件安全治理能力，为营造一个和谐、安全的网络环境贡献力量。

参考文献：

[1]　悬镜安全，ISC，中国电信研究院．软件供应链安全治理与运营白皮书[EB/OL]．[2023-05-04]．https://max.book118.com/html/2022/0827/5103210314004330.shtm.
[2]　蒋艳，赵冉，张格．国外开源软件安全治理模式研究及工作建议[J]．中国信息安全，2023(3)：76-79.
[3]　建信金科，中国信息通信研究院．软件物料清单(SBOM)安全应用白皮书[EB/OL]．[2023-05-04]．https://m.163.com/dy/article/HA5PFI0H0511A641.html.
[4]　董国伟．从美行政令看软件供应链安全标准体系的构建[J]．中国信息安全，2022(2)：84-87.
[5]　苏俐竹，徐雷，郭新海，等．国内外软件供应链安全现状分析与对策建议[J]．邮电设计技术，2022(9)：24-26.
[6]　中国信息通信研究院．开源生态白皮书[EB/OL]．[2023-05-04]．https://www.digitalelite.cn/h-pd-527.html?fromQz=false.
[7]　余建利，姜荣霞，卢蓉．电信运营商开源软件供应链安全治理探讨[J]．网络安全与数据治理，2023，42(1)：67-71，85
[8]　NIST. Survey of Existing Software Bill of Materials (SBOM) formats and standards[Z]. NISTIR(2022). 8321A.
[9]　KAPITSAKI G M，KRAMER F. Open source License violation check for SPDX files[C]//Software Reuse for Dynamic Systems in the Cloud and Beyond. Cham：Springer，2014：90-105.
[10]　KAPITSAKI G M，KRAMER F，TSELIKAS N D. Automating the License compatibility process in open source software with SPDX[J]. Journal of Systems and Software，2017，131：386-401.
[11]　吴江伟．软件供应链安全及防护工具研究[J]．中国信息安全，2021(10)：47-50.

作者简介：
王戈，工程师，硕士，主要从事网络与信息安全研究工作；郭新海，工程师，硕士，主要从事网络与信息安全研究工作；刘安，工程师，硕士，主要从事网络与信息安全研究工作；丁攀，工程师，硕士，主要从事网络与信息安全研究工作；蓝鑫冲，工程师，硕士，主要从事网络与信息安全研究工作。