ICS 35.040
CCS L 80

DB50

重　庆　市　地　方　标　准

DB50/T 1809—2025

软件供应链安全技术评价指南

2025－07－07 发布　　2025－10－07 实施

重庆市市场监督管理局　发布

目　　次

前　　言

本文件按照 GB/T 1.1—2020《标准化工作导则　第 1 部分：标准化文件的结构和起草规则》的规定起草。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由中共重庆市委网络安全和信息化委员会办公室提出、归口并组织实施。

本文件起草单位：数盾奇安（重庆）科技有限公司、重庆市质量和标准化研究院、工业和信息化部网络安全产业发展中心（工业和信息化部信息中心）、中共重庆市委网络安全和信息化委员会办公室、重庆市地矿测绘院有限公司、重庆西算大数据有限公司、重庆市互联网新闻研究中心、重庆若可网络安全测评技术有限公司、中兴通讯股份有限公司、重庆市璧山区党政信息中心、重庆兴农融资担保集团有限公司、先进操作系统创新中心（天津）有限公司、重庆依企莱科技发展有限公司、重庆数字城市科技有限公司、西南大学、重庆邮电大学、重庆理工大学、重庆交通大学、重庆中医药学院、重庆倍得林企业管理咨询有限公司、重庆恒扬禾信息技术有限公司、中国汽车工程研究院股份有限公司、中国烟草总公司重庆市公司、重庆三峡银行股份有限公司、马上消费金融股份有限公司、重庆奇安信科技有限公司、重庆市中冉数字科技有限公司。

本文件主要起草人：陈震宇、张彬哲、魏振国、郭威、李吉音、赵东、高二金、张波、毛艳、李蒙、马渊、李坪芮、王旭钢、王婷、陶永沛、何秋跃、胡涛、姜敏、黄建洪、高巍、王欢欢、陈雨阳、殷玲玲、杨先赟、雷剑梅、范开伟、宋海燕、刘家鑫、缪如燕、蒋洪志、韩熙、李宁、王悠悠、舒坤贤、刘颖、米波、卢军、唐明、杨秀峰、田进、冯欣、周洋、张力、陈阔、马培月、颜洁、王慧维、王梦洲、陈青扬。

软件供应链安全技术评价指南

1 范围

本文件提供了软件供应链安全技术评价工作的指导，给出了安全技术评价的概述、准备阶段、方案编制阶段、实施阶段和报告总结阶段的相关信息。

本文件适用于软件规划设计、开发实施及其运行维护过程中的安全技术评价工作，可为第三方机构开展软件供应链安全技术测评或测试提供依据。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 25069 信息安全技术 术语

GB/T 34943 C/C＋＋语言源代码漏洞测试规范

GB/T 34944 Java 语言源代码漏洞测试规范

GB/T 34946 C#语言源代码漏洞测试规范

3 术语和定义

GB/T 25069、GB/T 34943、GB/T 34944、GB/T 34946 界定的以及下列术语和定义适用于本文件。

3.1

软件供应链安全技术评价 software supply chain security evaluation

评价专业机构根据评价委托单位的评价目标、评价范围等，按照国家与地方法律法规及有关标准，应用风险框架，对软件生命周期过程中面临的各类安全风险，进行测试评估的活动。

注：本文件所指的软件供应链安全风险主要包括编码过程、开源组件、工具等技术性风险。

3.2

评价对象 target of evaluation

软件供应链安全评价工作涉及的业务软件、APP 应用等。

3.3

评价模块 module of evaluation

评价对象所包含的自建源代码、开源组件及扩展安全等具体评价工作模块。

4 缩略语

下列缩略语适用于本文件。

API：应用程序编程接口（Application Programming Interface）

APP：移动互联网应用程序（Application）
BSD：伯克利软件套件（Berkeley Software Distribution）
GPL：通用公共授权（General Public License）
LGPL：较宽松公共许可证（Lesser General Public License）
SBOM：软件物料清单（Software Bill of Materials）
XSS：跨站脚本（Cross-Site Scripting）

5 概述

5.1 评价过程

包括评价准备、方案编制、评价实施和报告总结等四个阶段。技术评价过程宜根据评价专业机构与评价委托单位之间协商结果，调整相应工作阶段及各阶段的任务。各阶段主要工作任务和详细工作流程见附录 A。

5.2 风险框架

在软件供应链各个供应活动中均可能引入安全隐患，导致软件供应链存在符合性风险、技术风险与管理风险。如图 1 所示，软件供应链符合性风险主要包括软件供应链的长期支持、知识产权、信创工程等国家及各行业、各领域共同关注的风险；软件供应链技术风险可能会进一步导致软件漏洞、软件后门、恶意篡改、信息泄露等安全风险；软件供应链管理风险主要关注流程、管理、人员、供应商、环境等安全风险。其他风险见附录 B。

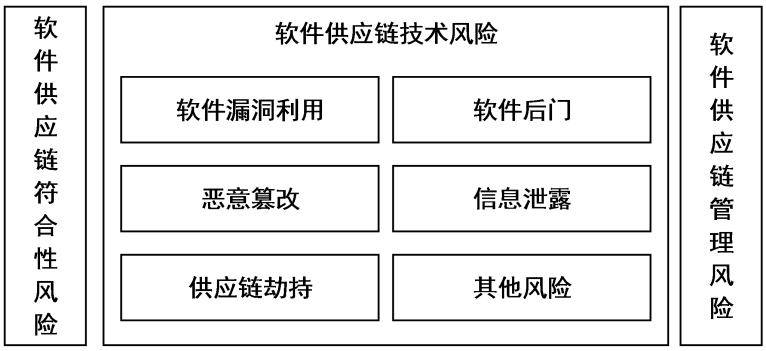

图 1 软件供应链风险框架

5.3 评价模块

包括源代码安全评价、开源组件安全评价和扩展安全评价（见图 2）。源代码安全评价针对自开发软件编写代码安全进行评价（见附录 C.2）。开源组件安全评价针对开源平台、开源组件进行评价（见附录 C.3）。扩展安全评价针对其他必要情形进行评价（见附录 C.4）。各评价模块与风险关系（见附录 C.5）。

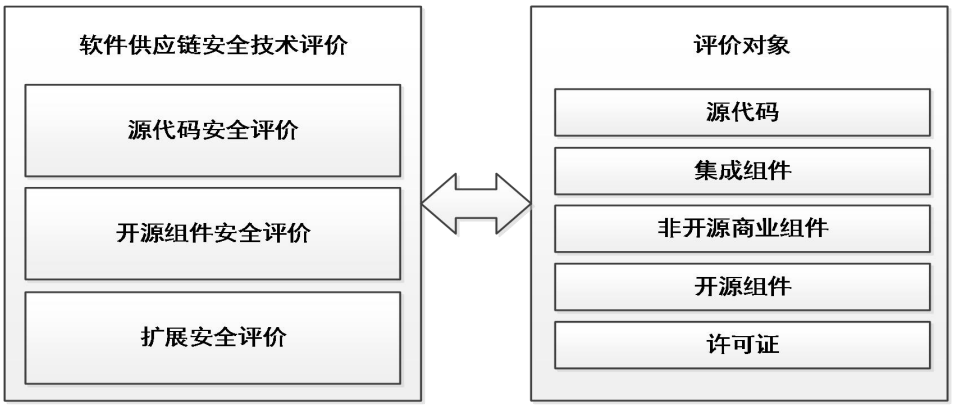

图 2　软件供应链安全技术评价模块

5.4　评价工作风险规避

在软件供应链安全技术评价工作中，可能会引发业务中断、业务服务能力下降、敏感信息泄露、商业秘密泄露以及恶意代码植入等危害。宜通过采取如下措施规避上述风险：

a) 签署评价项目授权协议。在评价工作正式开始前，宜签署评价项目授权协议，明确评价工作的目标、范围、人员安排、工作内容、工作流程、工作时间等；
b) 签署安全保密协议。根据评价工作接触的信息，相关机构及人员宜签署安全保密协议，约束评价过程信息采集、传输、存储、使用、发布等行为；
c) 在线评价风险规避。评价机构宜采取足够的安全措施，保证评价委托单位的代码元数据、代码、业务数据及个人信息等的安全；
d) 现场评价风险规避。对于重大风险情形，评价双方宜联合制定应急预案；对于模糊测试、渗透测试等动态安全评价，宜配置环境一致的模拟系统，在模拟环境下完成评价工作；
e) 评价信息安全保护。评价涉及开源组件、开源模块、源代码等原始数据，如无特别约定，在工作完成后评价机构宜在评价委托单位监督下，按照规范流程清除数据。

6　评价准备阶段

6.1　一般条件

6.1.1　评价机构

评价机构宜满足以下条件：

a) 组建软件供应链安全技术评价项目工作组；
b) 准备被评价对象基本情况调查表格；
c) 向评价委托单位相关责任人介绍软件供应链安全技术评价方法及流程；
d) 向评价委托单位说明软件供应链安全技术评价过程中可能存在的风险和规避方法；
e) 初步分析被评价对象的安全状况；
f) 准备用于评价工作的平台与工具。

6.1.2　评价委托单位

评价委托单位宜满足以下条件：

a) 为评价机构提供评价授权；
b) 向评价机构介绍被评价对象的基本情况；
c) 提供评价机构需要的相关材料；
d) 为评价人员的信息收集工作提供支持和协调；
e) 填写被评价对象的基本情况调查表；
f) 根据被评价对象的具体情况，为评价时间和评价工作安排适宜的建议；
g) 牵头制定应急预案。

6.2 工作过程

6.2.1 目标

宜包括以下内容：

a) 确保项目在预定时间内顺利开始；
b) 建立有效的项目沟通机制；
c) 明确项目的具体范围和界限；
d) 初步确定项目关键信息、平台与工具。

6.2.2 流程

评价准备阶段包括评价工作启动、范围与目标确定、信息收集和分析和平台与工具准备等四个主要任务，工作流程见图 3。

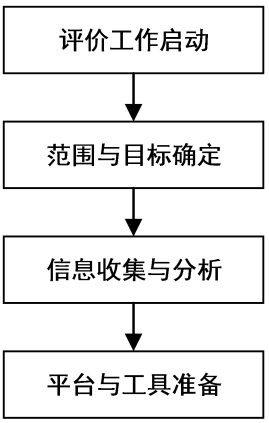

图 3 评价准备阶段工作流程

6.3 主要工作任务

6.3.1 评价工作启动

由评价机构与评价委托单位双方项目负责人、技术专家等相关人员举行项目启动会。主要工作任务宜包括：

a) 评价机构根据评价项目授权协议中的项目范围、项目目标和项目规模，组建评价项目工作团队，并编制项目计划书；
b) 评价机构对评价委托单位的项目干系人员，进行项目流程、评价手段、评价指标的详细讲解，并对评价委托单位在项目中的配合工作开展简单培训。

6.3.2 范围与目标确定

包括评价范围确定与评价目标确定，评价范围明确评价工作的对象范围、模块类型及规模，评价目标确定评价工作的价值。主要工作任务宜包括：

a) 根据约定范围，评价机构按照《范围与目标确认表》（见附录 D）收集并分析软件、APP 应用、开发平台、源代码、开源组件等基本信息；
b) 由评价委托单位确定整体评价目标，评价机构根据整体评价目标，编制各评价模块目标，各评价模块目标界限宜在整体评价目标界限内；
c) 评价双方根据收集的基本信息，结合项目进度、目标、风险等因素，确定最终评价范围。

6.3.3 信息收集与分析

评价机构根据确定的评价目标和评价范围，详细收集相应信息，进一步明确工作内容的组成，主要工作任务宜包括：

a) 评价机构明确评价对象信息，包括业务软件、APP 等使用情况、开发及维护文档等信息；
b) 评价委托单位根据评价机构提供的《开源组件及源代码调研表》（见附录 E），填写开源组件、源代码等信息；
c) 评价机构分析调研表格后评价工作难易程度，参考软件开发文档，形成《初步评价对象列表》（见附录 F）；
d) 评价委托单位确认《初步评价对象列表》。

6.3.4 平台与工具准备

宜采取自动化评价与人工验证相结合的方式，采用静态分析技术进行，评价方法主要为在线评价与现场评价。主要工作任务宜包括：

a) 评价机构依据初步确定的评价对象列表、开源组件列表、自建代码统计信息等确定评价方法；
b) 评价机构根据确定的评价方法，选择合适的评价平台、评价工具、评价辅助工具与提供配套文档等；
c) 评价机构就评价方法、评价平台使用中可能存在的风险进行提示，对评价委托单位相关人员开展评价平台、评价工具的使用与风险规避培训。

6.4 输入/输出文档

评价准备阶段主要输入与输出文档见表 1 。

表 1 评价准备阶段主要输入/输出文档

任务	输入文档	输出文档	主要内容
评价工作启动	评价项目授权协议	项目工作计划	项目目标、项目依据、工作步骤、评价内容、评价方法、进度及人员安排、技术重难点分析等
		会议纪要	会议时间、参会人员、主要事项、下一步工作计划等
范围与目标确定	评价工作启动输出 范围与目标调研表格 安全保密协议	评价范围清单	业务软件情况、APP 情况、源代码情况、开源组件情况等
		评价目标清单	整体目标、模块评价目标等
信息收集与分析	范围与目标确定输出 开源组件及源代码调研表格 软件开发文件	开源组件列表	维护组织、组件名称、开发语言、组件版本、用途等
		自建代码统计信息	开发语言、开发平台、开发工具等
		初步评价对象列表	业务软件、开源组件、源代码等关系列表

表 1　评价准备阶段主要输入/输出文档（续）

任务	输入文档	输出文档	主要内容
平台与工具准备	信息收集与分析输出 风险告知书	平台与工具列表	评价方法、评价平台、评价工具、评价辅助工具、评价涉及交接文档等
注：文档未包含应急响应相关文档，必要时，评价双方可参考 GB/T 24363—2009 第 6 章，编制应急预案。评价双方也可根据项目实际情况，增减相应的输入/输出文档，简化输出文档内容。			

7　方案编制阶段

7.1　一般条件

7.1.1　评价机构

评价机构宜满足以下条件：

a)　系统分析评价对象的组成、规模及工程难度；
b)　验证评价工作平台、评价相关工具的有效性、安全性；
c)　建议最终评价对象、评价模块、评价方法；
d)　编制包含风险规避计划的评价实施方案。

7.1.2　评价委托单位

评价委托单位宜满足以下条件：

a)　为评价机构分析提供资源支持；
b)　决定最终评价对象、评价模块、评价方法；
c)　审核与确定评价实施方案。

7.2　工作过程

7.2.1　目标

宜包括以下内容：

a)　界定评价的具体内容；
b)　确定评价方法；
c)　明确评价的重难点与流程；
d)　识别评价过程中潜在的风险点及规避方法。

7.2.2　流程

包括评价对象确定、评价模块确定、评价方法确定、评价方案编制和评价过程指南开发等五个主要任务，工作流程见图 4。

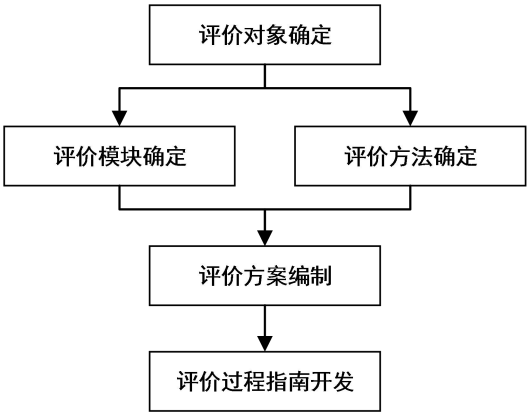

图 4　方案编制工作流程

7.3　主要工作任务

7.3.1　评价对象确定

根据《初步评价对象列表》，分析评价对象的规模、构成、特点，结合项目工作目标等因素，确定评价对象，并明确工作边界。主要工作任务宜包括：

a)　部署模拟环境或开发、发布环境提取方式，获取《初步评价对象列表》涉及评价对象的业务环境、开源组件、自建代码等信息；
b)　分析初步评价对象列表的自建代码规模、开源组件情况；
c)　分析各类因素，确定评价对象及其评价工作边界。

7.3.2　评价模块确定

根据评价对象确定自建源代码、开源组件及扩展安全等工作模块，主要工作任务宜包括：

a)　按照统一规则、唯一标识自建源代码，可根据软件、APP 应用对自建代码初步标识；
b)　按照识别的开源组件名称、版本号等信息标识开源组件；
c)　进一步确定自建代码中的开源组件、插件等信息；
d)　确定扩展评价模块信息。

7.3.3　评价方法确定

宜采用静态分析技术对源代码、开源组件等进行评价评估。静态分析技术采用主要方法见附录 G。源代码评价宜尽可能获取更多代码信息，根据预先设定的漏洞特征、安全规则等评价缺陷。

开源组件评价可提取软件中开源组件的开源软件标识、维护组织、版本信息等特征，宜通过平台、工具或人工的方式分析开源组件特征，评价漏洞、开源组件版本、开源许可协议等风险信息，也可直接提取开源组件源代码包、远程代码仓库发起开源软件分析任务。

评价双方宜沟通、验证方法的充分性、有效性、适宜性。主要工作任务宜包括：

a)　确定是否完全采用静态分析技术，如涉及动态分析技术，宜制定相应工作规范、应急预案等；
b)　确定采用的静态分析技术方法；
c)　确定在线评价方法与现场评价方法各自的适用范围。

7.3.4　评价方案编制

宜明确评价实施对象、模块、流程、方法、内容和风险规避等信息，编制大纲见附录 H。评价方案编制工作要点如下：

a) 根据确定的评价模块、评价方法，结合双方协商风险模型（见附录 B）及安全评价要素（见附录 C），确定评价内容；
b) 估算在线评价、现场评价、报告编制及其配套后续活动工作难度和工作量；
c) 进一步完善评价团队专业人员配置；
d) 编制评价工作计划，包括：人员安排、进度安排、评价内容安排等；
e) 编制风险规避计划；
f) 双方共同评审 a）～ e）内容，汇总形成评价实施方案。

7.3.5 评价过程指南开发

宜结合评价实施流程、风险规避计划编写，宜直观、详实及具有可操作性。主要工作任务宜包括：
a) 根据评价目标、评价模块、评价方法等，确定具体评价单元；
b) 验证评价模块、评价单元划分的合理性、可实施性；
c) 开发评价结果记录表单模板；
d) 培训相关人员。

7.4 输入/输出文档

方案编制阶段输入与输出文档见表 2。

表 2 方案编制阶段主要输入/输出文档

任务	输入文档	输出文档	主要内容
评价对象确定	初步评价对象列表	评价对象列表	待评价软件、APP 等列表
评价模块确定	/	评价模块名单	自建代码应用或 APP 名单、涉及开源组件名单、扩展评价清单等
评价方法确定	/	评价方法列表	自动化/人工评价、静/动态分析技术、在线/现场评价等
		评价模块、方法对应关系	评价方法、评价模块、评价单元
评价方案编制	评价范围清单 评价目标清单	评价方案	见附录 H
评价过程指南开发	各类标准规范	评价过程指南	评价模块、评价单元、单元评价目标、单元评价流程、单元预期结果等
注：本表的输入文档内容中，省略了上一阶段或上一任务输出的内容描述。			

8 评价实施阶段

8.1 一般条件

8.1.1 评价机构

评价机构宜满足以下条件：
a) 实施安全评价，并记录、分析结果；
b) 按照约定保护源代码、开源组件等安全，并在评价完成后按要求处置相关资源；
c) 必要时，启动并实施应急预案；
d) 以正式会议形式，向评价委托单位汇报评价初步结论。

8.1.2 评价委托单位

评价委托单位宜满足以下条件：

a) 提供评价环境（搭建模拟环境）、评价资源（如最新代码等）及评价配合人员；
b) 监督、配合评价实施与剩余信息保护（剩余信息保护指在处理敏感信息时，确保在处理完毕后，不会在系统中保留任何敏感信息），按需决定是否启动应急预案；
c) 确认评价初步结论。

8.2 工作过程

8.2.1 目标

宜包括以下内容：

a) 确保评价记录准确；
b) 确保评价过程安全；
c) 确保评价结果真实、有效。

8.2.2 流程

包括实施准备、评价执行和初步结论等三个主要任务，工作流程见图 5。

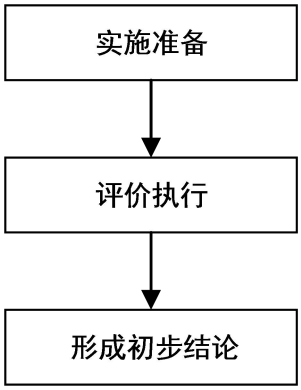

图 5 评价实施工作流程

8.3 主要工作任务

8.3.1 实施准备

根据评价模块、评价方法等，由评价双方精准定位评价对象、评价单元及进一步确定风险和规避方式。主要工作任务宜包括：

a) 召开评价实施会议，双方确认风险告知书，评价委托单位签发评价实施开工授权；
b) 评价机构搭建安全的、可靠的源代码评价环境，包括物理环境、软件编译环境等；
c) 评价委托机构搭建业务软件或 APP 应用模拟平台，授权提取源代码；
d) 双方确认评价环境、软件模拟环境、软件代码的真实性、有效性；
e) 进一步确定评价方案、评价过程指南的有效性。

8.3.2 评价执行

8.3.2.1 在线评价

通过互联网进行安全评价，主要用于开源组件类模块评价，在双方确认安全的情况下，也可用于源代码安全评价。主要工作任务宜包括：

a) 使用评价平台或评价辅助工具提取开源组件特征信息，并将特征信息上传到评价平台；
b) 根据评价实施方案、评价过程指南通过评价平台实施在线评价；
c) 根据双方约定处置开源特征码；
d) 在确保安全前提下，提取源代码，通过安全通道上传平台，对源代码实施在线评价，并按照约定处置源代码信息。

8.3.2.2 现场评价

在评价委托单位指定场所实施评价活动，在指定场所搭建评价平台，可用于所有模块实施评价。主要工作任务宜包括：

a) 搭建评价平台；
b) 测试评价平台有效性；
c) 根据评价实施方案、评价过程指南通过评价平台实施现场评价；
d) 根据双方约定处置开源特征码、源代码；
e) 必要时，通过动态评价技术进一步评价软件安全性。

8.3.3 形成初步结论

主要工作任务宜包括：

a) 双方确定评价工作结束，并签订评价结束确认书；
b) 汇总在线评价、现场评价所有评价结果；
c) 召开正式会议交换评价过程、初步结论信息。

8.4 输入/输出文档

评价实施阶段输入与输出文档见表3。

表 3 评价实施阶段主要输入/输出文档

任务	输入文档	输出文档	主要内容
实施准备	/	风险告知书	风险列表、可能后果等
		评价实施开工授权	授权评价源代码、开发平台、发布平台、开源组件等
评价执行	评价实施方案 评价过程指南	在线评价记录 现场评价记录	评价单元、评价单元记录、剩余信息处置记录等
形成初步结论	/	评价结束确认书	评价时间、参与人员、是否启动应急预案等
		初步风险列表	漏洞列表、版本建议、开发语言建议等
		会议材料	评价过程、评价初步结论、下一步工作计划等
注：本表的输入文档内容中，省略了如下信息： 1） 上一阶段或上一任务输出的内容描述； 2） 工作日志、会议纪要； 3） 非文档性输入/输出。			

9 报告总结阶段

9.1 一般条件

9.1.1 评价机构

评价机构宜满足以下条件：

a) 分析并评价评价单元、模块风险等级；
b) 关联分析并评价评价对象安全风险；
c) 编制包含安全改进建议的评价报告。

9.1.2 评价委托单位

评价委托单位宜满足以下条件：

a) 确认评价单元、模块、对象评价结果；
b) 审核并签收评价报告；
c) 召开总结会议，确认安全改进风险点。

9.2 工作过程

9.2.1 目标

宜包括以下内容：

a) 形成全面、准确的评价结论；
b) 汇报并沟通评价过程及结论。

9.2.2 流程

报告总结阶段包括模块风险判定、整体风险分析、评价报告编制和总结与建议等四个主要任务，工作流程见图 6。

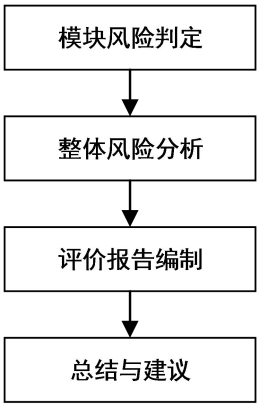

图 6 报告总结工作流程

9.3 主要工作任务

9.3.1 模块风险判定

根据模块的各单元评价结论，客观、准确地综合判定模块风险，并分析风险的活动。判定可全程人工完成，也可借助一定工具辅助。主要工作任务宜包括：

a) 根据模块，汇总评价单元记录、漏洞列表等信息，综合分析模块风险情况；
b) 双方确认各模块风险判定结果。

9.3.2 整体风险分析

风险分析过程可聘请有关专家参与。主要工作任务宜包括：

a) 对风险进行关联分析，识别其相互关系和影响，根据分析结果得出整体风险结论，并修正各模块和单元的高中低风险级别；
b) 确认风险之间依赖、耦合关系，对风险进行必要排序；
c) 必要时，根据风险分析结果及约定的评价依据（参考附录 I），给出评价结论。

9.3.3 评价报告编制

根据各类过程文档与评价结论，编制报告，报告格式见附录 J。主要工作任务宜包括：

a) 报告编制，编写详细的报告草稿，确保所有相关信息和数据完整且准确；
b) 组织报告评审，对评审意见进行整理和反馈，修改并完善报告内容；
c) 和评价委托单位确认报告内容，确保报告准确反映评价结论。确认无误后，正式发布最终报告。

9.3.4 总结与建议

根据评价过程与评价报告，总结评价工作，并对安全问题做简要改进建议，主要工作宜包括：

a) 以正式会议形式总结评价过程、评价结论，评价中的重点问题；
b) 对评价中必要的问题做改进建议：包括改进顺序、改进依赖、必须改进内容建议等；
c) 结束评价活动。

9.4 输入/输出文档

报告总结阶段输入与输出文档如见表 4。

表 4 报告总结阶段主要输入/输出文档

任务	输入文档	输出文档	主要内容
模块风险判定	初步风险列表	评价模块风险等级	模块风险名称、风险等级表等
		模块初步安全建议	模块风险后果简易说明、安全加固方法建议等
整体风险分析	/	关联分析记录	关联分析过程、关联分析结论等
		高风险分析情况	高风险类型、高风险列表等
		中低风险分析情况	中低风险类型、中低风险列表等
评价报告编制	/	评价报告	参见附录 J
总结与建议	风险准则	会议资料	评价报告内容等
		简易改进建议	风险排序、改进顺序建议等
注：本表的输入文档内容中，省略了如下信息： 1） 上一阶段或上一任务输出的内容描述； 2） 会议纪要。			

附 录 A
（资料性）
软件供应链安全技术评价活动流程

图 A.1 给出了软件供应链安全技术评价四个阶段涉及的活动流程。

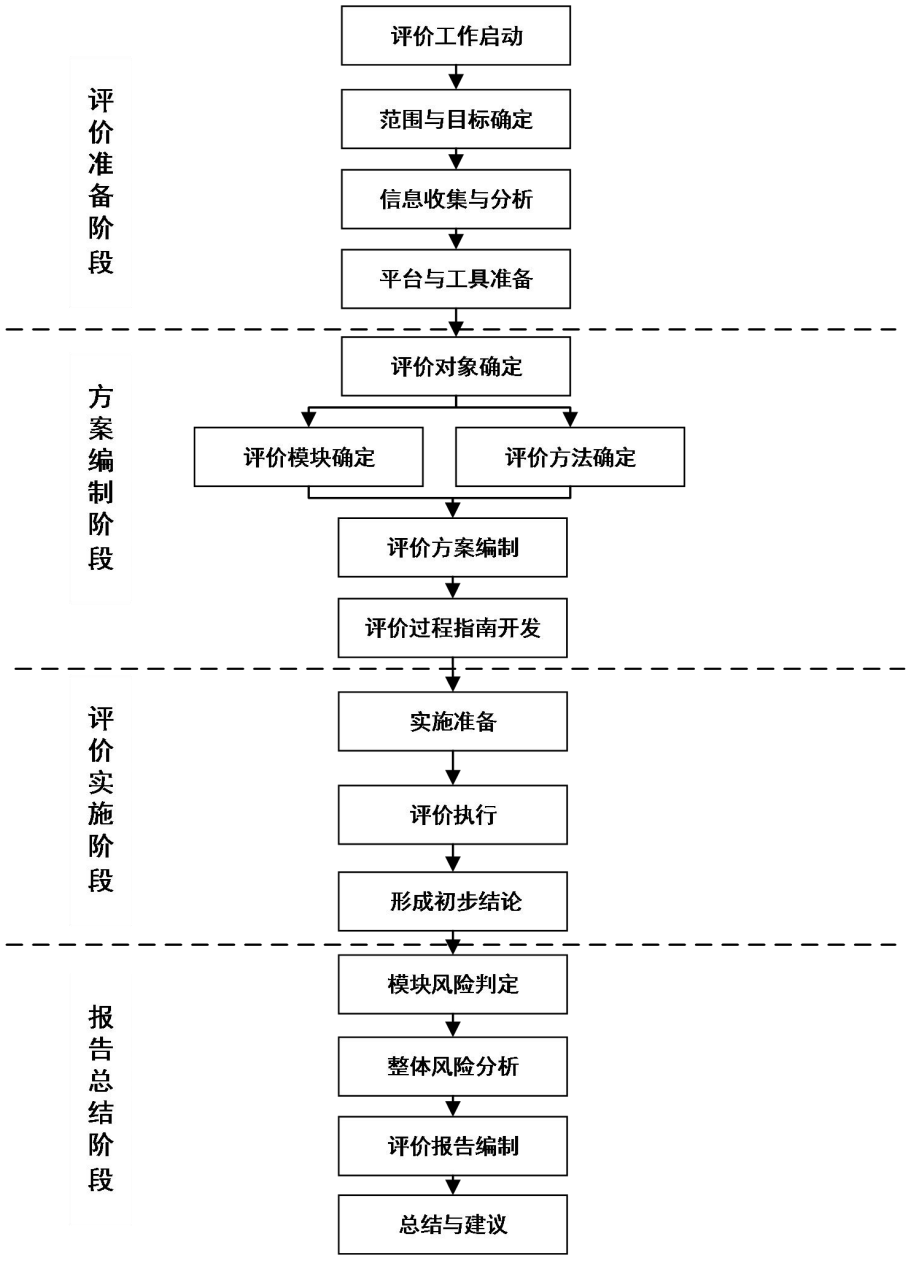

图 A.1 软件供应链安全技术评价活动流程

附 录 B
（资料性）
软件供应链技术安全风险

B.1 概述

主要包括软件漏洞、软件后门、恶意篡改、信息泄露和其他风险，为方案编制提供参考。

B.2 输入验证和表示

主要包括以下内容：

a) 缓冲区溢出。通过使用计算机向缓冲区内填充超过缓冲区本身的容量的数据位数，造成溢出数据覆盖在合法数据上。缓冲区溢出攻击能导致程序运行失败、系统宕机、重新启动等。攻击者甚至能利用缓冲区溢出执行非授权指令，获取系统特权，进而进行各种非法操作。
b) XSS。通过利用网页开发时留下的漏洞，将恶意指令代码注入应用程序的响应中，使用户加载并执行攻击者恶意制造的程序。这些恶意程序可能包括病毒、蠕虫、木马等，它们会损害用户的计算机和数据安全，或者执行一些未经授权的操作，如窃取用户信息、破坏系统文件等。
c) 格式化字符串。格式化字符串漏洞发生的原因通常是格式化字符串中的占位符与实际提供的参数不匹配。攻击者能通过构造包含特定格式说明符的输入字符串，来读取和修改程序内存中的敏感数据。格式化字符串漏洞的危害包括但不限于：信息泄露、代码执行、拒绝服务攻击。
d) 日志伪造。系统后台输出的日志中包含前端用户输入的内容时，用户能输入特定的符号或代码，从而篡改日志中原本应输出的正常内容。攻击者可能向日志文件中注入代码或其他命令，利用日志处理来执行恶意行为。
e) 路径操纵。攻击者能通过用户输入来控制文件系统操作所用的路径来访问或修改其他受保护的系统资源，达到攻击的目的。
f) SQL 注入。攻击者在应用程序的输入字段中插入或注入恶意的 SQL 代码，控制或操纵原本由应用程序执行的 SQL 查询，绕过应用程序的安全机制，直接与数据库进行交互。
g) 不安全的反射。攻击者使用具有反射功能的外部输入来选择不正确的类或代码，创建开发人员不想要的逻辑路径，这些路径可能会绕过身份验证或访问控制检查，导致系统执行未经授权的代码或命令、改变执行逻辑、崩溃、退出或重启。
h) XML 验证。XML 文件在处理过程中未经过充分或正确的验证，攻击者利用未经验证的 XML 输入来执行恶意操作，如代码注入、数据窃取或拒绝服务攻击等。

B.3 API 滥用

主要包括以下内容：

a) 危险函数。可能带来安全风险或潜在漏洞的函数，不当使用可能导致系统安全漏洞。
b) 堆检查。堆检查不当或未进行，导致内存逐渐积累，系统资源耗尽，程序性能下降，甚至系统崩溃。

B.4 安全特征

主要包括以下内容：

a) 不安全随机数。使用不安全的随机数生成算法或函数，带来的安全问题包括但不限于：易于猜测的密码、可预测的加密密钥、会话劫持攻击以及 DNS 欺骗等，可能导致敏感信息的泄露、未经授权的访问或系统资源的滥用等。
b) 最小权限。违反最小权限原则，攻击者能执行更高权限的操作，如读取或修改敏感数据、执行系统命令等。同时拥有过多权限的用户或进程可能访问和泄露不应公开的数据，带来的安全问题包括但不限于隐私泄露、商业机密泄露。
c) 密码管理。明文密码、空密码、硬编码密码、弱密码削弱了系统安全性，可能导致个人隐私信息泄漏和财产损失。
d) 配置管理。配置管理贯穿于整个软件生命周期，为软件研发提供一套管理办法和活动原则。不合规的配置管理可能导致敏感信息泄漏、重要数据丢失等危害。

B.5 代码质量

主要包括以下内容：

a) 双重释放。通过伪造整个内存块并欺骗操作系统，攻击者利用这种漏洞来修改内存内容，进而执行恶意代码或获取敏感信息。
b) 实现不一致。同一功能或特性在不同地方或不同版本的软件中有不同的实现方式或结果，带来的安全问题包括但不限于：错误和缺陷、安全漏洞、兼容性问题。
c) 内存泄漏。程序中已动态分配的堆内存由于某种原因未释放或无法释放，造成系统内存的浪费，导致程序运行速度减慢甚至系统崩溃等严重后果。
d) 空指针引用。攻击者能通过构造特定的输入或请求，触发程序中的空指针引用错误，使程序无法正常响应其他请求或提供服务，或利用这种错误来访问未授权的资源或篡改系统状态。
e) 未初始化变量。未初始化变量可能包含敏感信息，如内存中的残留数据。如果这些变量被错误地暴露，攻击者可能利用这些信息来进一步攻击系统或窃取数据。

B.6 恶意篡改

主要包括以下内容：

a) 恶意代码植入。在需方不知情的情况下，在软件产品或供应链中的组件中植入具有恶意逻辑的可执行文件、代码模块或代码片段。
b) 开发工具植入。使用被恶意篡改的开发工具，导致开发的软件或组件存在恶意代码。
c) 供应信息篡改。在供方不知情的情况下，篡改软件供应链上传递的供应信息，如销售信息、商品信息、软件构成信息等。

B.7 软件后门植入

主要包括以下内容：

a) 供方预留。供方出于软件维护的目的，在软件产品中预置后门，如果预置后门被泄露，攻击者会通过预置后门获得软件或操作系统的访问权限。
b) 攻击者恶意植入。攻击者入侵软件开发环境，污染软件供应链中的组件，劫持软件交付升级链路，攻击软件运行环境植入恶意后门，获得软件或操作系统的访问权限。

B.8 供应链劫持

供应链劫持是普遍存在的一种供应链污染，安全风险突出，涉及捆绑恶意代码、下载劫持、网络劫持、物流链劫持、升级劫持等。

B.9 其他风险

B.9.1 开源许可违规使用

主要包括以下内容：

a) 无开源许可证。软件产品发布时缺少开源许可证类型，包括但不限于 LGPL、GPL、BSD、Apache 等许可证。
b) 使用不规范。软件产品发布时不符合相应许可协议的规范和要求，包括但不限于没有遵循开源许可证协议。开源组件使用或修改过程中，存在许可证不合规、不兼容或丢失等风险。

B.9.2 假冒伪劣

供方提供未经产品认证、评价的软件或组件，或未按照声明和承诺提供合格的产品。

B.9.3 供应中断

主要包括以下内容：

a) 突发事件中断。因自然等不可抗力、政治、外交、国际经贸等原因造成上游软件、使用许可、知识产权授权的中断。
b) 不正当竞争。软件供方利用需方对产品和服务的依赖，实施不正当竞争或损害用户利益的行为。

B.9.4 源代码管理

主要包括以下内容：

a) 版本控制。利用源代码文件不同版本的差异性，进行恶意包装，实现对源代码的控制。
b) 源代码权限管理。对源代码文件的权限或者安全进行威胁，从而实现源代码泄露或者利用。
c) 源代码来源管理。利用开源代码规模占比、源代码编码语言占比情况，实现对源代码的来源管理。
d) 源代码质量管理。利用源代码漏洞率、源代码漏洞严重性、源代码漏洞影响程度、源代码漏洞利用复杂程度、源代码漏洞修复率、源代码版本更新情况，实现对源代码的质量管理。
e) 源代码知识产权管理。利用开源许可证规范性、开源许可证互惠性、开源许可证兼容性、开源许可证专利及授权情况、开源许可证适用范围，实现对源代码知识产权管理。

B.9.5 开发安全管理

主要包括以下内容：

a) 软件包元数据篡改风险。攻击者可能篡改软件包的元数据，误导用户安装含有恶意代码的软件包，对用户的系统和数据安全造成潜在危害。
b) 包管理工具安全风险。使用不受信任的包源或不进行包的安全检查和验证，无法保障包管理工具的安全性，可能导致软件引入潜在的安全漏洞。
c) 容器镜像安全风险。容器镜像中可能包含已知漏洞或恶意组件，导致数据泄露、服务中断等风险。

d) CI/CD 集成环境安全风险。若 CI/CD 管道未得到妥善保护，攻击者可能在编译、测试、发布阶段注入恶意代码，对软件开发和发布流程构成严重威胁。
e) 代码托管平台风险。选择不受信任的代码托管平台或未实施严格的访问控制和安全审计，可能导致代码和资产遭受未经授权的访问和篡改，进而引发安全风险。
f) 云平台安全风险。使用云平台时，若云平台本身存在安全漏洞或受到攻击，这些漏洞和攻击可能直接影响软件的安全性，导致软件遭受未经授权的访问和数据泄露等风险。
g) 软件下载过程风险。软件下载过程中若未采用安全协议（如 HTTPS），下载捆绑软件或恶意组件，无法保障软件的完整性和安全性。
h) 交付范围风险。随着交付范围的扩大，新增组件和功能可能未经充分的安全测试和验证，进而引入新的安全风险，对整体软件的安全性构成潜在威胁。

附 录 C
（资料性）
软件供应链安全技术评价要素

C.1 概述

本附录提供编制评价要素时的参考要素，为评价工作提供依据。

C.2 源代码安全技术评价要素

本部分不涉及特定语言的特定评价内容，仅提供评价的通用要素，实际评价中宜根据实际语言情况，进一步参考 GB/T 34943、GB/T 34944、GB/T 34946 等标准，对要素做适当的剪裁或扩充，最终形成组合的评价单元，表 C.1 给出了源代码安全评价要素所涉及的评价要素、要素内容、评价要点的内容。

表 C.1 源代码安全技术评价要素

评价要素	要素内容	评价要点
代码实现	开发语言	源代码程序所使用的语言类别、版本情况。
	代码仓库	源代码存储的代码仓库来源。
	配置环境	源代码运行环境的权限、属性、组件调用等。
代码质量	命名规则	使用有意义的变量、函数和类命名，以反映其用途和功能。
	注释规则	提供有益的注释和文档，解释代码的意图、实现细节和重要的决策。
	代码风格	采用一致的代码格式化规则和缩进风格，使代码具有统一的外观。
	逻辑单元	将代码块细分为较小的逻辑单元，避免过长的代码行和函数。
	模块规则	将代码分解为模块或类，每个模块或类仅负责一个清晰明确的任务。
	函数规则	函数和方法具有清晰的输入和输出，遵循良好的参数传递和返回值规范。
	函数安全	弃用不安全函数，使用与开发语言版本匹配的安全推荐函数。
	语法错误	能快速识别语法性错误，遵循良好编写规则，不会因语法错误导致错误的执行。
文件管理	路径安全	安全开发的意识、培训开展情况，每次项目都开展安全工程师语言特性的培训并记录； 发生软件供应链安全事件，及时开展总结培训。
	文件权限	在创建文件时候，根据文件敏感级别设置不同的访问权限，防止敏感数据被授权之外的读写。
	上传管理	按规则将上传文件重命名或严格评价文件名、文件路径的合规性。
	文件名管理	严格管理文件名中的特殊字符及文件后缀名。
内存操作	内存越界	使用异常评价等机制，及时发现或终止向前、向后的内存越界行为。
	地址校验	校验写入内存地址，防止任意地址写入。
加密解密	密码安全	密文存储密码等敏感信息，通过变换算法或者配置等方式设置密码或者密钥。
	加密强度	使用安全的对称加密、非对称加密、哈希算法，密钥强度符合要求； 使用编程语言推荐的加密强度较强的算法。

表 C.1　源代码安全技术评价要素（续）

评价要素	要素内容	评价要点
敏感数据保护	敏感数据输出	仅输出必要的最小数据集，避免多余字段引起敏感数据泄露； 有敏感信息输出脱敏措施； 敏感信息不在日志或其他渠道密文输出。
	敏感数据存储及访问	敏感数据加密存储； 敏感数据使用独立的存储层与严格的访问控制。
错误与异常	必要捕获异常	对程序影响巨大的异常，采取严格的异常捕获。
	自定义异常捕获	对重大逻辑问题，设置报错或异常捕捉机制。
	错误信息	分离管理与用户端错误信息，最小化提供错误信息。
身份认证	口令安全	口令采用大小写字母、数字及特殊字符的组合，密码强度为 8 位～16 位； 采取评价机制避免近几次使用重复密码。
	身份验证	只有经过身份认证和授权的用户才能访问系统或特定资源。
信息验证	输入验证	所有程序外部输入的参数值，进行数据校验。校验内容宜包括：数据长度、数据范围、数据类型与格式。校验不通过，拒绝。
	逻辑验证	代码对利用外部输入来构造命令或部分命令时，对其中的特殊元素进行了处理。
命令执行	白名单机制	尽量避免直接调用操作系统命令，优先使用 API 实施文件操作，必要的操作采取函数与系统双白名单机制实施。
	避免意外执行	禁止外部数据直接作为操作系统命令制定，并禁止通过创建 shell 后，拼接外部数据执行操作系统命令。
程序日志	日志记录	重要行为都记录日志，敏感信息不保存在日志中。
	日志存储	日志保存时限与业务目标一致。
软件后门	后门植入	检测软件后门，避免对系统进行未授权的访问和操作。
配置文件	审计监控	对配置文件进行审计和监控，检查是否存在未经授权的变更或潜在的安全风险。
	安全性保障	配置文件往往包含敏感信息，如数据库连接信息、API 密钥等。因此，需要采取加密、访问控制等安全措施，确保配置文件的安全性。

C.3　开源组件安全技术评价要素

表 C.2 给出了开源组件安全评价要素所涉及的评价要素、要素内容和评价要点的内容。

表 C.2　开源组件安全技术评价要素

评价要素	要素内容	评价要点
版本控制	版本历史	记录文件和代码的修改历史。
	分支和合并	支持开发团队创建分支来并行开发不同的功能或修复问题，同时分支可合并回主线（通常是主分支）以融合代码变更，确保代码的一致性和合并冲突的解决。
	回滚和恢复	开发者可回滚到先前的版本，以恢复到稳定状态或撤销错误的修改。
	访问控制	管理对代码库的访问权限，并为不同的用户或用户组设置不同的权限级别。
依赖管理	依赖声明	明确项目所需的依赖项，包括库、框架、工具和插件等。
	依赖解析	解析依赖项及其依赖树，从而确定每个依赖项所依赖的其他组件和版本。
	版本管理	确定每个依赖项所使用的版本，并处理依赖项之间的版本冲突。
	获取和安装	从依赖管理仓库或其他来源下载和安装所需的依赖项。

表 C.2 开源组件安全技术评价要素（续）

评价要素	要素内容	评价要点
依赖管理	升级管理	严格测试后，按需更新依赖项可获取最新的功能、修复漏洞或安全问题，并与其他组件保持兼容。缓存已下载的依赖项，以便在多个项目或环境中重复使用。
	冲突解决	不同的依赖项要求使用不同版本的相同库或组件时，能处理依赖项之间的冲突。
	漏洞管理	评价依赖项的安全漏洞和依赖组件的可信度。
源代码成分	组件成分	使用依赖分析、Hash 匹配、文件解析等技术识别代码中的开源组件成分。
	文件签名组件	将源代码工程的完整目录结构进行签名计算之后，与开源项目的文件签名库进行对比，评价开源组件。
	代码片段组件	通过将源代码中的代码片段进行特征提取之后，与开源项目的代码片段进行对比，评价开源组件。
	代码片段漏洞	通过分析源代码中代码片段与漏洞影响代码的相似性，从而识别源代码中的漏洞信息。
	代码克隆	通过提取源代码工程中完整的代码片段特征与开源项目或对比项目的代码片段进行对比，从而计算出被测代码的开源率、同源率信息。
	文本内容协议	通过分析源代码中文件中包含的开源许可协议声明内容以及 License 等文件，综合识别出源代码中包含的开源许可协议。
	开源协议版权	通过分析源代码文本内容中涉及 CopyRight、版权符号等内容，识别被测对象的开源许可协议版权风险。
	SBOM 评价	解析 SBOM 并生成对应的结果文件进行分析，识别开源组件。
二进制成分	开源组件	二进制/可执行文件中的开源组件成分。
	容器镜像	容器镜像中开源组件成分，包含基础镜像中的开源组件、应用中的开源组件等。

C.4 扩展安全技术评价要素

表 C.3 给出了扩展安全评价要素所涉及的评价要素、要素内容和评价要点的内容。

表 C.3 扩展安全技术评价要素

评价要素	要素内容	评价要点
安全管理	安全意识和培训	安全开发的意识、培训开展情况，每次项目开展安全工程师语言特性的培训并记录；发生软件供应链安全事件，及时开展总结培训。
	应急响应	建立紧急响应计划和应急响应流程，每年一次应急演练。
源代码来源	开源代码规模	软件产品包含的各开源代码模块字节数在软件产品代码中所占比例。
	源代码编码语言	源代码所使用的编码语言种类在软件产品代码中所占比例。
源代码质量	源代码漏洞率	源代码原始漏洞数量和千行漏洞率情况。
	源代码漏洞严重性	源代码原始漏洞严重性（指标项参考 GB/T 30279—2020 中 6.3.3 要求）。
	源代码漏洞影响程度	源代码原始漏洞影响范围（指标项可参考 GB/T 30279—2020 中 6.2.2 中影响程度要求）。
	源代码漏洞利用复杂	源代码原始漏洞攻击复杂性（指标项可参考 GB/T 30279—2020 中 6.2.1 中影响程度要求）。
	源代码漏洞修复率	源代码漏洞修复率及修复时间。
	源代码版本更新情况	源代码版本更新频率、最新版本情况、版本维护情况。

表 C.3 扩展安全技术评价要素（续）

评价要素	要素内容	评价要点
源代码合规	开源许可证规范性	软件产品开源许可证规范性情况：授权范围、授权条件、违约与授权终止、免责声明等。
	开源许可证互惠性	源代码是否存在互惠性开源许可证，对自由互惠的开源许可证进行识别和风险评估。
	开源许可证兼容性	软件产品包含的开源许可证兼容性情况，判断源代码许可证是否兼容合规。
	开源许可证专利情况	源代码涉及的专利是否得到授予。
	开源许可证适用范围	软件产品包含的开源许可证的适用范围和出现纠纷时法律声明情况。

C.5 技术评价要素与风险对应关系

表 C.4 给出附录 C.2 ～ C.4 评价要素与附录 B 的简略对应关系。

表 C.4 评价要素与风险对应关系

评价要素	要素内容	评价要点
源代码评价	代码实现	B.2 输入验证与表示 B.3 API 滥用 B.4 安全特征 B.7 软件后门植入
	代码质量	B.5 代码质量
	文件管理	B.4 安全特征 d）配置管理 B.9.4 源代码管理
	内存操作	B.2 输入验证与表示 a）缓冲区溢出 B.5 代码质量 c）内存泄漏
	加密解密	B.2 安全特征 g）密码管理
	敏感数据保护	B.2 输入验证与表示 B.3 API 滥用 B.4 安全特征 B.5 代码质量
	错误与异常	B.2 输入验证与表示 B.3 API 滥用 B.4 安全特征 B.5 代码质量
	身份认证	B.2 安全特征 g）密码管理
	信息验证	B.2 输入验证与表示
	命令执行	B.3 API 滥用
	程序日志	B.2 输入验证与表示 d）日志伪造 B.4 安全特征 d）配置管理
开源组件评价	版本控制	B.9.4 源代码管理 a）版本控制
	依赖管理	B.8 供应链劫持
	源代码成分	B.6 恶意篡改 a）恶意代码植入
	二进制成分	B.6 恶意篡改 b）开发工具植入
扩展评价	安全管理	/
	源代码来源	B.9.4 源代码管理 c）源代码来源管理
	源代码质量	B.9.4 源代码管理 d）源代码质量
	源代码合规	B.9.4 源代码管理 b）源代码权限管理 e）源代码知识产权

附 录 D
（规范性）
范围与目标确认表

表 D.1 给出了范围与目标确认表。

表 D.1 范围与目标确认表

评价委托单位			
事项负责人		联系方式	
评价机构			
事项负责人		联系方式	
评价范围			
对象范围	□业务软件 □APP □开发平台 □开发工具 □其他		
模块范围	□源代码 □开源组件 □扩展安全		
软件/APP	业务软件（业务软件名称、版本、部署位置等）		
	APP（APP 名称、版本、部署位置等）		
开发平台/工具	开发平台		
	开发框架		
	版本控制系统		
	CI/CD 工具		
评价目标			
安全评价目标	识别并评估供应链中的安全漏洞和弱点	□是	□否
	确保所有组件符合行业安全标准与法规要求	□是	□否
	提升软件供应链的透明度和可追溯性	□是	□否
	建立或优化软件供应链安全管理体系	□是	□否
	其他特定目标	如有其他特定需求请填写	
备注	本表可根据实际情况删减、扩展或添加附件。		

附 录 E
（规范性）
开源组件及源代码调研表

E.1 开源组件调研表

表 E.1 给出了开源组件调研表示例。

表 E.1 开源组件调研表

评价委托单位					
事项负责人		联系方式			
评价机构					
事项负责人		联系方式			
开源组件信息					
组件名称	版本号	用途描述	许可证类型	下载地址或仓库链接	关联业务软件/APP
备注	本表可根据实际情况删减、扩展或添加附件。				

E.2 源代码调研表

表 E.2 给出了源代码调研表示例。

表 E.2 源代码调研表

评价委托单位			
事项负责人		联系方式	
评价机构			
事项负责人		联系方式	

表 E.2　源代码调研表（续）

源代码信息					
业务软件/APP	代码行数	主要编程语言	代码结构概述 （如模块、功能组织）	安全实践 （现有安全开发实践或代码审查机制）	安全措施 （如输入验证、代码审查）
备注	本表可根据实际情况删减、扩展或添加附件。				

附 录 F
（资料性）
初步评价对象列表

F.1 概述

本附录提供了初步评价对象列表的示例，帮助评价委托单位和评价机构明确评价的范围和对象。

F.2 基本信息

基本信息宜包括：

a） 项目名称；
b） 委托单位；
c） 项目负责人；
d） 联系方式；
e） 评价日期范围。

F.3 评价对象概述

F.3.1 业务软件/APP信息

业务软件/APP 信息宜包括：

a） 名称；
b） 版本号；
c） 描述（简要说明软件的功能和用途）；
d） 运行环境（如操作系统、浏览器兼容性等）；
e） 部署位置（本地、云服务等）。

F.3.2 开发及维护文档

开发及维护文档宜包括：

a） 是否可用：是/否；
b） 文档类型（如设计文档、架构图、API 文档、用户手册等）；
c） 文档链接/存放位置。

F.3.3 开发平台/工具

开发平台/工具宜包括：

a） 主要编程语言；
b） 开发框架；
c） 版本控制系统（如 Git、SVN 等）；
d） CI/CD 工具（如 Jenkins、GitLab CI 等）。

F.4 源代码与组件信息

F.4.1 源代码

源代码信息宜包括：

a) 是否开源：是/否；
b) 源代码库地址（如 GitHub、GitLab 等）。

F.4.2 第三方开源组件

开源组件列表（按照表格形式填写，宜包括组件名称、版本、许可证类型、关联系统等）。

F.4.3 安全评价目标

安全评价目标根据实际调研结果填写。

F.5 评价委托单位确认

评价委托单位确认宜包括：

a) 确认人；
b) 确认日期；
c) 备注（如有需要特别说明的事项）。

附 录 G
（资料性）
主要源代码静态分析技术

G.1 概述

静态分析技术采用的方法主要包括规则检查、约束分析、数据流分析、符号执行、内存精确建模和控制流分析等。

G.2 规则检查

程序本身的安全性可由安全规则描述。程序本身存在一些编程规则，即一些通用的安全规则，也称之为漏洞模式。规则检查方法将这些规则以特定语法描述，由规则处理器接收，并将其转换为分析器能接受的内部表示，然后再将程序行为进行比对、评价。

G.3 约束分析

将程序分析过程分为约束产生和约束求解两个阶段，前者利用约束产生规则建立变量类型或分析状态之间的约束系统，后者对这些约束系统进行求解。

约束系统可分为等式约束、集合约束和混合约束三种形式。等式约束的约束项之间只存在相等关系，集合约束把每个程序变量看成一个值集，变量赋值被解释为集合表达式之间的包含关系，混合约束系统由部分等式约束和部分集合约束组成。

G.4 数据流分析

数据流分析技术是一种在不运行程序的条件下，从程序中获取数据流信息的技术，数据流信息最终被传递给缺陷评价分析引擎进行进一步缺陷分析。在数据流信息获取方面，分析的精度问题至关重要，主要从流不敏感、流敏感和路径敏感三个方面来增加分析的精度。

G.5 符号执行

假设程序的所有输入值都是符号值，根据程序中的每条路径对程序进行符号模拟执行，在程序分支处，记录程序对变量的约束信息，同时求解约束条件，判断该条路径是否可执行，从而可剪除不可执行路径。采用这种方式有如下优点：

——最大可能地评价了程序的所有路径；

——避免了虚假路径带来的误报问题。

G.6 内存精确建模

通过对内存建模，可对指针表达式的值进行精确分析，并区分指向同一个对象内部的指针各自不同的偏移量，使得针对指针的评价更加精确。

G.7 控制流分析

对程序执行时可能经过的路径进行图形化表示，根据不同语句之间的关系，尤其是由“条件转移”“循环”等引入的分支关系，通过对这些语句的处理，得到关于程序结构的图形化分析，可评价潜在危险的操作执行顺序。

附 录 H
（资料性）
软件供应链安全评价方案

H.1 概述

本附录给出了软件供应链安全评价方案简要示例，包含正文和附件的要素信息。

H.2 正文

H.2.1 项目概述

项目概述包含但不限于：

a） 项目背景；
b） 项目目标；
c） 项目依据；
d） 项目范围。

H.2.2 评价对象信息

评价对象信息宜包括：

a） 软件或 APP 情况；
b） 开源组件清单；
c） 自建代码统计；
d） 上次评价主要问题；
e） 上次评价整改情况。

H.2.3 评价流程

评价流程主要包括评价涉及主要工作，宜包括：

a） 评价实施阶段主要工作；
b） 报告总结阶段主要工作；
c） 主要输出物。

H.2.4 评价方法

评价方法包括如下方法的一种或多种：

a） 在线评价；
b） 现场评价。

H.2.5 评价内容

可参考附录 C 的内容编制评价内容，包括：

a） 源代码安全评价；
b） 开源组件安全评价；

c） 扩展安全评价。

H.2.6 评价项目管理

评价项目管理宜包括：

a） 评价平台与工具；
b） 评价进度计划；
c） 评价人员安排；
d） 评价风险规避；
e） 评价配合资源要求，指对委托机构的人员、代码等要求。

H.3 附件

附件宜包括：

a） 过程输入、输出的表单模板；
b） 源代码评价示例；
c） 开源组件评价示例。

附 录 I
（资料性）
评价依据

I.1 概述

本附录给出了评价依据，帮助评价单位出具最终评价结论。

I.2 判定依据

评价结论根据附录 C 或约定的评价内容，由评价初步结论与风险分析结论共同决定。评价结论包括通过、基本通过、不通过三个等级（若评价依据特定的规范性文件，结论宜为符合、基本符合、不符合）。具体判定依据见表 I.1。

表 I.1 评价结论判定依据

评价结论	判定依据
通过	初步结论通过率高于 80%，且风险分析无高风险。
基本通过	初步结论通过率高于 60%，低于 80%，且风险分析无高风险； 或初步结论通过率高于 80%，且存在不高于 5 个高风险点。
不通过	初步结论通过率低于 60%，或存在高于 5 个高风险点。

附 录 J
（资料性）
软件供应链安全评价报告

J.1 概述

本附录给出了软件供应链安全评价报告简要示例，包含封面、正文及附件的要素信息。

J.2 封面

封面要素包括但不限于：

a) 报告编号：可包含年、月及其评价双方单位编号等信息；
b) 标题信息：软件供应链安全评价报告；
c) 其他内容：委托单位、评价机构、出具报告时间（到月）。

J.3 正文

J.3.1 评价结果摘要

结果摘要内容要素包含但不限于：

a) 评价时间简要说明；
b) 评价对象简要说明；
c) 评价依据；
d) 评价结果与问题简要说明。

J.3.2 评价对象情况

对象情况包含但不限于：

a) 业务软件情况；
b) 软件供应链模块基本情况；
c) 开源组件清单；
d) 自建代码统计。

J.3.3 安全评价过程

安全评价过程宜包括：

a) 总体评价流程；
b) 评价方法说明；
c) 评价内容说明。

J.3.4 源代码安全评价

源代码安全评价宜包括：

a) 源代码评价结果；

b） 源代码安全分析；
c） 源代码安全小结。

J.3.5 开源组件安全评价

开源组件安全评价宜包括：
a） 开源评价结果；
b） 开源组件安全分析；
c） 开源组件安全小结。

J.3.6 扩展安全评价

扩展安全评价宜包括：
a） 扩展评价结果；
b） 扩展评价部分安全分析；
c） 扩展安全小结。

J.3.7 整体安全分析

整体安全分析宜包括：
a） 安全风险分类汇总；
b） 高风险脆弱性分析；
c） 中风险脆弱性分析；
d） 低风险脆弱性分析。

J.3.8 结论与建议

整体安全分析宜包括：
a） 安全结论；
b） 安全脆弱性改进建议。

J.4 附件

附件内容宜包括：
a） 安全评价过程必要的过程记录、截图等信息；
b） 软件供应链安全技术风险列表；
c） 软件安全缺陷高、中、低风险分类规则；
d） 评价委托单位风险准则（如果有）；
e） 评价机构与人员信息。

参 考 文 献

[1] GB/T 28449 信息安全技术 网络安全等级保护测评过程指南

[2] GB/T 30279—2020 信息安全技术 网络安全漏洞分类分级指南

[3] GB/T 37970 软件过程及制品可信度评估

[4] GB/T 38674 信息安全技术 应用软件安全编程指南

[5] GB/T 43698 网络安全技术 软件供应链安全要求

[6] YD/T 3447 联网软件源代码安全审计规范

[7] ISO/IEC TR 24772-1:2019 Programming languages. Guidance to avoiding vulnerabilities in programming languages—Language-independent guidance

[8] NIST SP 800-218 Secure Software Development Framework(SSDF)Version 1.1 ： Recommendations for Mitigating the Risk of Software Vulnerabilities

[9] 国家网络空间安全战略（2016 年 12 月 27 日国家互联网信息办公室发布）

[10] 网络安全审查办法（2021 年 11 月 16 日国家互联网信息办公室 2021 年第 20 次室务会议审议通过）

[11] “十四五”软件和信息技术服务业发展规划（工信部规〔2021〕180 号）