ICS 35.030
CCS L80

中华人民共和国国家标准

GB/T XXXXX—XXXX

信息安全技术 移动互联网应用程序（App）生命周期安全管理指南

Information security technology—Guidelines for life cycle security management of mobile internet applications（App）

(点击此处添加与国际标准一致性程度的标识)

（征求意见稿）

（本稿完成时间：2022 年 1 月 26 日）

在提交反馈意见时，请将您知道的相关专利连同支持性文件一并附上。

XXXX－XX－XX 发布 XXXX－XX－XX 实施

国家市场监督管理总局
国家标准化管理委员会 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。

本文件由全国信息安全标准化技术委员会（SAC/TC 260）提出并归口。

本文件起草单位：武汉安天信息技术有限责任公司、华为技术有限公司、北京赛西科技发展有限责任公司、维沃移动通信有限公司、三六零科技集团有限公司、OPPO广东移动通信有限公司、北京小米移动软件有限公司、公安部第三研究所、国家计算机病毒应急处理中心、中国软件评测中心、国家计算机网络应急技术处理协调中心、中国科学院信息工程研究所、启明星辰信息技术集团股份有限公司、联想（北京）有限公司、海信集团有限公司、蚂蚁科技集团股份有限公司、南方电网数字电网研究院有限公司、北京智游网安科技有限公司（爱加密）、深圳市腾讯计算机系统有限公司、杭州安恒信息技术股份有限公司、北京指掌易科技有限公司、北京百度网讯科技有限公司、北京版信通技术有限公司。

本文件主要起草人：潘宣辰、许玉娜、陈诚、衣强、成明江、姚一楠、李腾、陆伟、水晶、张艳、刘彦、蔡一鸣、孙海燕、何能强、杜丹、史景、李汝鑫、张湋易、王昕、母天石、陈家林、韩云、刘海涛、李献振、李彪、吴月升、董宏、陈牧、潘正泰、王克、方宁。

引　　言

在移动互联网技术高速发展的浪潮下，移动互联网应用程序出现在人们衣食住行的各个场景，生活与移动互联网已经变得形影不离。但是，开放生态下的App面临着恶意程序、安全漏洞和侵害用户权益等复杂的风险，App开发者与移动应用分发平台厂商和移动智能终端厂商进行着复杂的交互。

本文件提出了开发者、移动应用分发平台厂商和移动智能终端厂商对移动互联网应用程序生命周期进行协同管理以降低上述风险，不同角色可以根据实际工作参考使用本文件。本文件的制定是为了让开发者在开展规范性实践时有据可依、加强认识，从而促进整个移动互联网生态的健康发展。

信息安全技术　移动互联网应用程序（App）生命周期安全管理指南

1　范围

本文件提供了移动互联网应用程序（App）生命周期安全管理的建议。

本文件适用于App开发者对App的开发、运营，也适用于移动应用分发平台厂商和移动智能终端厂商对App的管理，也可作为第三方机构对App进行安全检测时的参考。

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 35273—2020　信息安全技术　个人信息安全规范

3　术语和定义

GB/T 25069和GB/T 35273—2020界定的以及下列术语和定义适用于本文件。

3.1

移动智能终端 mobile intelligent terminal

具有能提供应用程序开发接口的开放系统，并能安装和运行第三方应用软件的移动终端。

[来源：GB/T 39720—2020，3.1]

3.2

移动互联网应用程序 mobile Internet application

安装在移动智能终端设备上，能够利用移动智能终端设备上操作系统提供的开发接口，实现某项或某几项特定任务的计算机程序。

注：包括移动智能终端预置、下载安装的应用程序和小程序，简称App。

[来源：GB/T 34978—2017，3.1.2，有修改]

3.3

App 生命周期 mobile Internet application life cycle

App从需求分析到终止运营随时间进化的过程。

3.4

App 开发者 mobile Internet application developer

执行开发活动并对开发完成的App承担责任的主体。

注：包括法人、自然人或者其他组织等，简称开发者。

3.5

移动应用分发平台 mobile application distribution platform

App下载或升级的渠道。

注：包括应用商店、App、网站、游戏中心和短视频等。

3.6

小程序 mini program

基于应用程序开放接口实现的，用户无需安装即可使用的App。

注：应用程序通过公开应用程序编程接口（API）或函数，使外部的程序可以增加该应用程序的功能或使用该应用程序的资源，而不需要更改该应用程序的源代码。

4 缩略语

下列缩略语适用于本文件。

API：App编程接口（Application Programming Interface）

SDK：软件开发工具包（Software Development Kit）

5 概述

5.1 App 生命周期安全需求

App的生命周期主要包含七个阶段：需求分析阶段、开发设计阶段、测试验证阶段、上架发布阶段、安装运行阶段、更新维护阶段和终止运营阶段，七个阶段都可能面临各种不同的安全风险，需要在App生命周期中对每个阶段进行安全分析与安全管理，见图1 App生命周期安全保证框架示意图。

注：App生命周期可以帮助移动互联网生态中不同参与者（如开发者、移动应用分发平台厂商和移动智能终端厂商）沟通和理解各个阶段的活动。当App有多个版本的迭代开发或管理时，也宜参考该App生命周期。

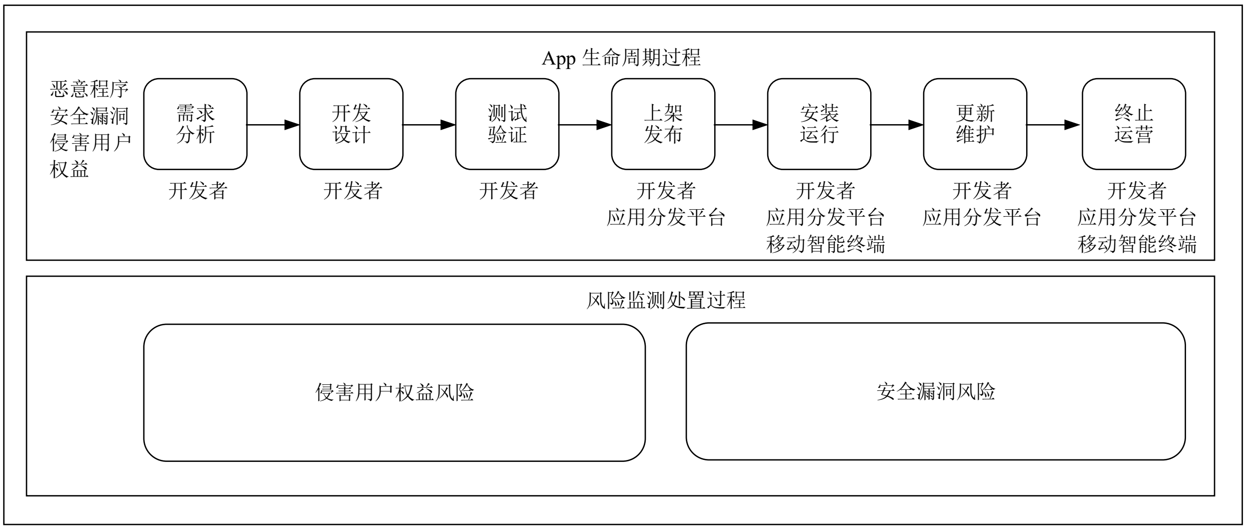

图1 App 生命周期安全保证框架示意图

5.2 App 生命周期安全保证框架

根据App在生命周期的开发、运营和管理过程中的不同特点，App可能面临的安全风险（见附录A）包括但不限于：

a） App自身可能会面临被恶意程序攻击的风险，恶意程序的分类及风险描述见表A.1；

b） 当App存在侵害用户权益的问题时，会面临被通报或者下架的风险，侵害用户权益的分类及风险描述见表A.2；

c） App存在的安全漏洞可能会被攻击者利用的风险，影响App的正确运行，安全漏洞的分类及风险描述见表A.3。

App生命周期安全保证框架由App生命周期过程和风险监测处置过程组成。在App生命周期过程中，各阶段开展不同的管理或技术活动，应对可能出现的风险，降低出现风险的可能性，安全风险应对体系见附录B。风险监测处置过程是通过采集生命周期过程中部分阶段的行为数据，并对行为数据进行风险分析和特征运营，从而判定App是否存在侵害用户权益风险，最终进行风险处置，降低App被通报或者下架的可能性；通过采取措施防范安全漏洞信息的泄露，防止安全漏洞信息传播而产生危害，降低App面临安全漏洞被利用的可能性。

5.3 角色及安全建议

不同角色的安全建议包括App开发时的自身安全和生命周期管理时的协同安全两个方面，具体如下：

a) 开发者在开发App时，宜参考App的程序安全和安全保障相关的建议，见附录C。程序安全包括：编码安全、通信安全、访问控制、日志记录与保护和数据保护与密码等方面的内容；安全保障包括：环境安全、第三方SDK或组件安全、发布安全和个人信息安全等方面的内容。

b) App生命周期不同阶段的安全管理，由开发者、移动应用分发平台厂商和移动智能终端厂商协同来实现。

   1) 开发者主要参与 App 的需求分析、开发设计、测试验证、更新维护等阶段，通过开展需求分析及评审、安全质量保证、方案设计、方案评审、代码管理、变更控制、安全测试、安全验证、安全交付、安全维护和安全更新等方面的安全管理活动，有助于减少 App 发生安全漏洞被利用和侵害用户权益等问题；
   2) 移动应用分发平台厂商主要参与 App 的上架发布阶段，通过开展开发者身份资质及信誉管理、上架审核、App 管理和整改监督等方面的安全管理活动，有助于减少 App 存在的虚假身份信息、开发者信誉不良和安全检测被绕过等问题；
   3) 移动智能终端厂商主要参与 App 的安装运行和终止运营等阶段，通过开展安装检测、运行状态检测和终止运营等方面安全管理管理活动，有助于降低 App 面临的恶意程序、安全漏洞和侵害用户权益等安全风险。

6 生命周期管理

6.1 生命周期过程

6.1.1 需求分析阶段

6.1.1.1 需求分析及评审

开发者宜参考以下与需求分析及评审相关的管理活动。

a) 在程序设计前进行安全需求分析，根据法律法规、标准约束和客户安全需求等形成安全需求说明书。

b) 安全需求说明书中主要分析的内容包括但不限于：

   - App所支撑的业务特点；
   - 威胁、脆弱性、业务风险场景分析；
   - 安全需求与保护目标；
   - 个人信息安全相关；
   - App宜提供的安全功能；

- App内的权限控制措施；
- 关键操作的抗抵赖功能；
- App与其他App或系统之间的数据交互保护，以及需要保护用户的哪些数据。

c) 基于系统环境和业务安全等安全需求开展前期的风险评估，识别潜在的安全风险，形成风险分析报告。

d) 制定整体安全策略，编制安全措施报告，包括安全目标和总体安全措施等内容。

e) 组织风险管理部门与安全技术团队参与评审，对安全需求说明书、风险分析报告和安全措施报告进行评审，评审内容包括但不限于以下方面：

- 身份认证及授权；
- 访问控制；
- 安全审计；
- 数据生命周期安全；
- App容错。

6.1.1.2 安全质量保证

开发者宜参考以下与安全质量保证相关的管理活动：

a) 制定安全质量目标，为每个安全质量度量项制定目标水平，度量项包括可接受的漏洞等级和数量、需访问的业务相关敏感权限的最小个数、完整性指标、保密性指标和可用性指标等；

b) 制定安全质量保证计划，包括开发过程中开展的安全开发、安全测试和安全交付等安全活动，只发布达到安全质量目标的计划；

c) 编制安全质量文档，包括安全质量目标和安全质量保证计划等内容，并组织质量保证工程师和开发人员对安全质量文档进行评审；

d) 制定安全质量管理监控机制，当安全质量目标无法实现时，采取适当的防范措施；

e) 执行安全质量保证过程活动和任务计划，在App生命周期内执行和维护该计划，App需要符合所制定的标准和规程。

6.1.2 开发设计阶段

6.1.2.1 安全开发

开发者宜参考附录C开展安全开发相关的技术活动。

6.1.2.2 方案设计

开发者宜参考以下与方案设计相关的管理活动。

a) 根据安全需求说明书进行安全设计，进行威胁建模和攻击面分析，形成安全设计方案，内容包括但不限于以下方面：

- 对App的默认攻击面和最大攻击面进行安全分析，使攻击面最小化；
- 采用威胁建模等方法对App业务进行安全分析，并对分析出来的安全风险制定消减方案；
- 对有商业用途的开源软件进行安全风险评估，如开源软件中是否包含已知安全漏洞、开源软件是否一直处于维护更新状态等；
- 对个人信息进行安全评估，并对个人信息采取相应的安全保障措施。

b) 编制安全开发方案，包括开发过程中涉及的安全技术等内容。

6.1.2.3 方案评审

开发者宜参考以下与方案评审相关的管理活动：

a） 制定评价准则，包括安全需求的可追溯性、与安全需求的一致性、可测试性、安全设计方案的可行性、运行和维护的可行性等；
b） 组织相关方安全设计方案和安全开发方案进行评审；
c） 记录评审结果并形成评审记录。

6.1.2.4 代码管理

开发者宜参考以下与代码管理相关的管理活动：
a） 制定安全编码规范，并在开发过程中严格执行；
b） 不在代码中嵌入有风险的代码，如无关链接、逻辑炸弹、后门、恶意程序或者侵害用户权益的代码等；
c） 编码完成后使用代码扫描工具和人工代码检查方式进行评审，识别安全缺陷并修复；
d） 对源代码的访问进行权限控制；
e） 对程序资源库进行修改、更新或发布等操作时需要经过授权和批准；
f） 当代码由供应商提供时，编码完成后将代码回收并防止泄密。

6.1.2.5 变更控制

开发者宜参考以下与变更控制相关的管理活动：
a） 明确变更需求，根据变更需求制定变更方案，仅在通过评审和审批后实施变更方案；
b） 对方案或代码的变更需进行权限控制，通过评审和审批后提交；
c） 记录变更操作，以便进行安全审计和回退操作；
d） 当开发环境变更时，考虑对App开发的影响，做好变更前的环境备份以便恢复。

6.1.3 测试验证阶段

6.1.3.1 安全测试

开发者宜参考以下与安全测试相关的管理活动：
a） 建立安全测试机制，编制安全测试设计文档，描述安全测试方法，包括恶意程序检测、安全漏洞检测、违法违规行为与内容检测等；
b） 编制安全测试用例文档，记录输入的实际值与预期的输出内容；
c） 当使用测试工具或人工测试时，先制定测试验收指标，形成测试技术方案，并得到授权后方可进行测试；
d） 按照安全测试设计文档和项目计划中的安全检查点设计执行安全测试，形成安全测试总结报告，并对安全测试总结报告进行评审；
e） 将测试环境与系统开发环境、生产环境隔离，不宜使用开发环境或生产环境作为测试环境，不宜将测试环境转为生产环境；
f） 当App由供应商开发时，对供应商开发交付的App进行安全测试和评估；
g） 当存在未在隐私政策中说明的用户个人信息收集、处理、传输、存储等行为时，检测该行为是否符合个人信息保护相关政策法规。

6.1.3.2 安全验证

开发者宜参考以下与安全验证相关的管理活动：
a） 根据已确定的验证任务，制定验证计划并形成文档；
b） 验证设计和编码正确地实现了安全功能、安全保密和其他关键性的安全需求；
c） 验证App的部件或单元已完整且正确地集成到App中；

d) 在实施验证计划过程中，解决发现的问题和不符合项，并将解决的问题和不符合项以及对应的措施进行记录和文档化管理，形成经验教训。

6.1.3.3 安全交付

开发者宜参考以下与安全交付相关的管理活动：

a) 建立 App 交付验收流程和制度，并在新建、升级和更新版本时进行交付验收；
b) 提供安全维护的指导性文档，给出适当的风险提示和应急响应措施，明确安全的部署环境；
c) 根据安全需求说明书和安全质量文档对安全功能进行验证和风险评估；
d) 完成对 App 安全性测试过程中发现的漏洞、病毒、木马等有害文件的修复和处理；
e) 对 App 中存在的关于侵犯用户个人信息安全或侵害用户权益的行为进行整改和消除；
f) 通过可靠渠道交付，提供验证所交付 App 完整性的安全措施，减少交付过程中的篡改风险。

6.1.4 上架发布阶段

6.1.4.1 开发者身份资质及信誉管理

移动应用分发平台厂商对开发者进行身份资质和信誉管理，有助于减少App存在的虚假身份信息和开发者信誉不良等问题。移动应用分发平台厂商宜参考以下相关的管理活动：

a) 认证开发者真实身份，以便 App 上架后可以追溯到相应的真实开发者；
b) 审核开发者上架申请 App 的著作权，检查开发者是否具有相关的 App 著作权或著作权人的授权；
c) 建立开发者信誉评价管理体系，并定期对开发者信誉进行评估，优先选择信誉良好的开发者；
d) 当开发者与移动应用分发平台厂商联合运营 App 时，信誉评价结果欠佳的开发者不宜参与安全相关的运行运营。

6.1.4.2 上架审核

移动应用分发平台厂商宜参考以下与上架审核相关的管理活动：

a) 审核 App 安装包的完整性，不上架在发布过程中被篡改的 App；
b) 审核 App 的来源可靠性，不上架被二次开发而存在违法违规行为的 App，如私自破解、汉化、反编译、重新打包和换皮等，如果特定场景中的定制化需要重新打包上架，提供相关方的授权；
c) 审核 App 的代码，不上架存在恶意程序或安全漏洞等安全风险的 App；
d) 审核 App 的行为，不上架存在侵犯用户个人信息安全或者侵害用户权益等问题的 App；
e) 审核 App 的其他行为，不上架存在开发者身份信息不真实或联系方式虚假失效等问题的 App。

6.1.4.3 App 管理

移动应用分发平台厂商宜参考以下与App管理相关的活动：

a) 建立使用 App 安全管理制度，如安全风险管理、安全事件管理、应急响应管理、个人信息管理等；
b) 宜对已发布的 App 安全状况进行监测，如被盗版、被仿冒等；
c) 宜对已发布的 App 在运行时所受到的攻击和安全风险进行定期感知、告警和处置；
d) 宜对承载 App 业务相关的数据库进行登记，并跟踪该数据库的安全状况；
e) 向用户明示开发者在 App 提交时声明其获取的用户终端权限及用途；
f) 建立完善用户举报投诉的反馈渠道，并在合理时间内对问题进行响应。

6.1.4.4 整改监督

App在整改监督上的安全管理由开发者和应用分发平台协同进行：

——移动应用分发平台厂商宜参考以下与整改监督相关的管理活动：

- 检测出存在违法违规行为的App，对开发者进行告知；
- 督促存在违法违规行为的App进行整改，包括但不限于存在恶意程序、安全漏洞、侵害用户权益等安全风险的App；
- 对检测出存在安全漏洞问题的App，宜立即告知开发者，并采取措施防止漏洞信息泄密；
- 对检测出存在侵犯用户个人信息安全或侵害用户权益问题的App给出整改建议，对于未按期限整改的，依据相关要求/规定对App进行下架处理；
- 对整改后通过安全检测和行为检查，仅对符合相关要求/规定的App进行恢复上架处理。

——开发者宜参考以下与整改相关的管理活动：

- 存在侵害用户权益问题的App，整改措施宜参考6.2.1.3；
- 存在安全漏洞问题的App，整改措施宜参考6.2.2.3 和6.2.2.4。

6.1.5 安装运行阶段

6.1.5.1 安装检测

移动智能终端厂商宜参考以下与安装检测相关的管理活动：

a) 检测 App 是否包含恶意程序、安全漏洞和侵害用户权益等风险；

b) 将检测信息展示给用户，包括但不限于检测对象和结果等；

c) 若检测过程中识别出安全风险，进一步展示安全风险的详细说明和处置建议等；

d) 根据白名单控制 App 的安装、运行。

6.1.5.2 运行状态检测

App在运行状态上的安全管理由开发者、移动智能终端、应用分发平台协同进行：

——开发者厂商自行检测 App 在运行状态下是否存在恶意程序、安全漏洞或侵害用户权益等风险；

——移动智能终端为用户提供 App 监测的触发；

——应用分发平台厂商督促开发者对运行中的安全问题进行整改。

6.1.6 更新维护阶段

6.1.6.1 安全维护

App在维护上的安全管理由开发者和应用分发平台协同进行：

——开发者宜参考以下与安全维护相关的管理活动：

- 采集App运行过程中产生的日志，宜采用管理平台集中管理；
- 不在日志中记录未经匿名化处理的个人信息；
- 保护日志记录，并设置日志保存时间；
- 当由第三方提供运行与维护服务时，与第三方签订服务水平和保密协议。

——应用分发平台在维护上的安全管理宜参考 6.1.4.3 中的 a）、d）和 e)。

6.1.6.2 安全更新

App在更新上的安全管理由开发者和应用分发平台协同进行：

——开发者宜参考以下与安全维护相关的管理活动：

- 更新前，校验客户端完整性；

- 更新前，对更新安装包进行恶意程序、安全漏洞和侵害用户权益等风险检测，并评估安全风险；
- 更新前，对待更新的功能进行可行性测试，并记录测试结果；
- 更新前，告知用户更新的内容和版本变更情况等；
- 采取安全措施保护升级过程中的用户信息；
- 对更新过程中启用安全机制，不通过热更新或热补丁等方式更新App，不在更新后引入新的安全风险。

——应用分发平台在更新上的安全管理宜参考 6.1.4.3 中的 b）和 c)。

6.1.7 终止运营阶段

App在终止运营阶段的安全管理由开发者、应用分发平台和移动智能终端协同进行：

——开发者宜参考以下与终止运营相关的管理活动：

- 明确告知用户App将终止运营；
- 停止收集和使用个人信息，删除已经保存的个人信息或进行去标识化处理；
- 进行风险评估，确保不引入新的风险；
- 对终止运营的App及其相关文档进行归档，包括但不限于开发文档、日志和编码等文档，并设置对归档数据副本的访问权限；
- 在不影响移动智能终端安全使用的情况下，附属的资源文件、配置文件和用户数据文件等也能够被方便卸载，包括但不限于用户个人信息或者重要数据不储存到公共区域；
- 在用户触发卸载行为后，客户端残留数据中不包含个人信息或者重要数据。

——应用分发平台厂商宜参考以下与终止运营相关的管理活动：

- App在终止运营后，进行下架处理；
- 对严重违规的App，按相关规定/要求下架整改处置。

——移动智能终端厂商宜参考以下与终止运营相关的管理活动：

- 不恢复删除后的数据；
- 对严重违规的App，按相关规定/要求中止其运行。

6.1.8 其他支持过程

6.1.8.1 安全管理制度

开发者宜参考以下与安全管理制度相关的活动：

a) 建立符合组织整体安全策略的安全管理制度，实现对 App 生命周期安全管理；
b) 建立 App 安全风险管理机制，包括安全风险评估、安全风险处置和安全应急响应等；
c) 建立 App 安全审计管理机制，包括安全事件的日期和时间、用户、事件类型、事件是否成功及其他与审计相关的信息等，并对发生的安全事件进行回溯和责任定位；
d) 建立并遵守源代码安全管理制度。

6.1.8.2 人员管理

开发者宜参考以下与人员管理相关的活动：

a) 根据组织业务对 App 的依赖性、对处理业务的保密性、可用性和完整性需求，设置适当的安全管理组织和岗位，并定义角色分工；
b) 对设计和开发人员进行安全技术培训，执行安全开发方法和生命周期管理措施，并进行安全审计；

c） 对技术人员进行安全基本概念和安全基础知识培训，如常见安全漏洞介绍、检测手段、漏洞原理以及危害、编码安全和代码安全性审计方法等；
d） 对相关人员进行安全意识培训，如办公安全、物理环境安全、移动智能终端安全以及生活中的信息安全等；
e） 对参加培训的人员进行安全技能评价。

6.1.8.3 文档管理

开发者宜参考以下与文档管理相关的活动：

a） 对生命周期各阶段的产出的安全文档进行配置管理，如安全需求说明书、安全措施报告 、风险分析报告、安全质量保证计划 、安全质量文档、安全开发方案、安全编码规范 、变更方案、安全测试设计文档、安全测试用例文档、安全测试总结报告和安全维护的指导性文档等；
b） 维护开发项目组的人员清单，并设置不同人员对不同文档的访问权限，并详细记录；
c） 严格管理需求分析阶段收集到的业务、用户和系统文档资料，宜设置专人管理；
d） 加密存储重要的文档资料和包含用户个人敏感信息的文档等；
e） 验证设计和开发文档的输入数据的来源和适合性，宜采用自动化的文档编辑工具；
f） 参考设计和开发文档相关的编制标准中规范文档的格式、技术内容和表述方式，评审和编辑所编制的文档，并通过授权人批准后发布文档；
g） 不通过携带纸质文档、邮件、网盘或移动存储等方式泄露重要文档。

6.1.8.4 内容管理

开发者宜参考以下与内容管理相关的活动：

a） 建立内容安全检查机制，对含有违法信息、不良信息的内容进行识别和删除；
b） 建立内容安全管理机制，对发布违法违规信息内容的 App 采取处置措施，保存记录并向有关监管部门报告；
c） 不在 App 中通过文字、音/视频等方式误导、欺骗、诱导用户达到不正当目的。

6.1.8.5 漏洞管理

开发者宜参考以下与漏洞管理相关的活动：

a） 当发现 App 存在安全漏洞后，采取措施对安全漏洞进行验证，评估安全漏洞的危害程度和影响范围；
b） 向监管部门报送已发现漏洞的相关信息；
c） 及时对已发现的安全漏洞进行修复。

6.1.8.6 业务数据管理

除了参考各类业务数据的安全标准，开发者宜参考以下与业务数据管理相关的活动：

a） 根据业务场景建立数据安全能力，降低业务数据被篡改、泄漏、损毁或丢失等风险；
b） 执行业务数据保护相关的管理和技术措施。

6.1.8.7 个人信息管理

开发者宜参考以下与个人信息管理相关的活动。

a） 收集个人信息后，宜进行去标识化处理，将可用于恢复识别个人的信息与去标识化后的信息分开存储并加强访问和使用的权限管理。

b) 根据用户授权情况，设置个人信息存储的最短期限，法律法规另有约定或者用户另行授权同意的除外，超出上述个人信息存储期限后，需要对个人信息进行删除或匿名化处理。

c) 采用加密等方式传输和存储个人信息。

d) 个人生物识别信息和个人身份信息分开加密储存。

e) 对个人生物识别信息进行安全处理，处理方法包括但不限于：

- 仅存储个人生物识别信息的摘要信息；
- 在使用面部识别特征、指纹、掌纹、虹膜等实现识别身份、认证等功能后，删除可提取个人生物识别信息的原始图像。

f) 当共享个人信息时，需要事先征得用户的授权同意，开展个人信息安全影响评估，依评估结果采取有效的保护用户的措施，并为因共享发生的损害用户合法权益的安全事件负责。

g) 当通过界面展示个人信息时，宜对需要展示的个人信息进行去标识化处理。

h) 建立相应的内部制度和政策对员工提出个人信息保护的工作指引。

i) 对个人信息保护政策、相关规程和安全措施的有效性进行审计，审计过程形成的记录作为安全事件的处置、应急响应和事后调查等活动的参考依据。

6.2 风险监测处置过程

6.2.1 侵害用户权益风险

6.2.1.1 行为数据采集活动

6.2.1.1.1 行为数据采集

开发者采集App在生命周期中的行为数据，为行为风险分析提供依据，宜参考以下与行为数据采集相关的活动：

a) 在 App 生命周期中采集不同阶段的行为数据，如开发设计、测试验证、安装运行、更新维护、终止运营等阶段；

b) 在 App 获取操作系统权限时采集权限获取行为数据；

c) 在 App 运行时采集与业务操作相关的行为数据。

6.2.1.1.2 行为数据定义与分类

开发者宜参考以下与行为数据定义与分类相关的活动：

a) 按产生时间定义行为，包括 App 在开发、测试、安装、运行、卸载等时间产生的行为；

b) 按 App 获取的权限定义行为，包括获取通讯录、短信、通话记录、位置等操作系统权限；

c) 按 App 实现的业务定义行为，包括平台推送、支付、资金理财、广告推送等；

d) 对采集的行为数据进行分类和描述。

6.2.1.2 行为风险运营活动

6.2.1.2.1 行为风险特征提取

开发者对App存在的行为风险数据进行特征提取是自动化分析行为风险的有效手段，行为风险的特征类型宜参考表A.2。

6.2.1.2.2 行为特征运营

开发者宜参考以下与行为特征运营相关的活动：

a) 根据不同行为特征形式对行为数据进行分类管理；

a) 对不同类型的行为特征进行关联分析，维护行为特征模型，为行为风险判定提供模型支撑。

6.2.1.2.3 行为风险判定

开发者宜参考以下与行为风险判定相关的活动：

a) 在特定场景下根据多个行为特征数据的综合判定来识别行为风险；

b) 当判定结果在一个预置范围内而无法确定是否为行为风险时，可借助其他因素来判定，如该行为被投诉的频次等。

6.2.1.3 行为风险处置

开发者宜参考以下与行为风险处置相关的活动。

a) 立即制定问题整改的版本升级方案。

b) 发现属于 SDK 组件存在的行为风险时：

  - 立即告知相关产品的开发者，并协商对SDK的整改措施；
  - 完善SDK组件管理制度。

c) 在发布升级版本前进行充分严格的运行测试，确保原有风险问题已解决，且避免衍生问题。

d) 提供有效且便利的途径，让用户在安全的通道下获取升级版本。

e) 若开发者被告知 App 出现行为风险，除参考 a)～d) 的条款外，还需按照相关政策的处置要求对风险进行处置。

6.2.2 安全漏洞风险

6.2.2.1 安全漏洞的获知

6.2.2.1.1 安全漏洞的发现

App开发者或App运营者（以下简称“开发者”）在进行人工或者自动化方法进行漏洞的探测、分析活动时，宜参考以下内容：

a) 不对用户的系统运行和数据安全造成影响和损害，不为了发现漏洞而侵犯其他组织的业务运行和数据安全的行为；

b) 不得利用操作系统或其他应用中存在的漏洞而进行非预期操作；

c) 在识别网络产品或服务的潜在漏洞时，主动评估可能存在的安全风险；

d) 采取防止漏洞信息泄露的有效措施；

e) 宜建立应用安全漏洞奖励机制，对发现并通报所提供应用安全漏洞的组织或者个人给予奖励；

f) 不得利用操作系统中的漏洞进行开发，如果发现操作系统中存在漏洞，立即告知相关移动智能终端的操作系统厂商；

g) 如果发现属于其他产品或者组件存在的安全漏洞，立即告知相关产品的开发者。

6.2.2.1.2 安全漏洞信息的接收

开发者建立安全漏洞信息接收渠道并保持畅通，并采取措施保障漏洞信息的安全和保密接收，安全漏洞的接收活动宜参考以下内容：

a) 制定并发布漏洞信息接收策略，包括但不限于漏洞接收范围、漏洞接收渠道、漏洞接收要求、漏洞接收流程等内容；

b) 建立产品安全漏洞应急响应团队，并由专人负责；

c) 不以应用终止运营为由，拒绝接收漏洞信息；

d) 在收到漏洞信息后，及时给予漏洞报告者确认或反馈；

e) 留存应用安全漏洞信息接收日志不少于 6 个月；
f) 采取有效措施保证漏洞相关信息的安全和保密性，防止漏洞信息泄露；
g) 提供技术措施保证信息流转渠道安全。

6.2.2.2 安全漏洞的验证

开发者获知应用存在安全漏洞后，及时对漏洞的存在性、等级、类别（参考附录A.3 ）等进行技术验证，评估安全漏洞的危害程度和影响范围。安全漏洞的验证活动宜参考以下内容：

a) 如果被报告的漏洞是在目前不提供支持的应用中发现的，继续完成调查和漏洞验证，并确认该漏洞对其他支持的应用的影响；
b) 在漏洞被证实后，根据漏洞验证情况并依据 GB/T 28458—2020 中 5.3 的要求对漏洞进行描述，并及时将漏洞信息报送给主管部门的漏洞信息共享平台，报送内容包括但不限于存在应用安全漏洞的产品名称、型号、版本以及漏洞的技术特点、危害和影响范围等。

6.2.2.3 安全漏洞的修复及防范措施

在验证漏洞的真实性后，开发者及时组织对应用安全漏洞展开修补活动。安全漏洞的修补活动宜参考以下内容：

a) 对已确认的漏洞，在考虑漏洞严重程度、受影响用户的范围、被利用的潜在影响等因素的基础上，立即进行漏洞修复，或制定漏洞修复及防范措施；
b) 对于需要用户采取软件、固件升级等措施的安全漏洞，及时将应用安全漏洞风险及修补方式告知可能受影响的产品用户，并提供必要的技术支持完成漏洞修复；
c) 在发布补丁和升级版本前进行充分严格的有效性和安全性测试，避免补丁衍生的应用功能和安全缺陷，对于不能通过补丁或版本升级解决的漏洞风险，提出有效的临时处置建议，出具技术指导说明；
d) 对于依据 GB/T 30279-2020 中 6.3.3 评定的技术分级为超危、高危的漏洞，若不能立即给出修复措施，给出有效的临时防护建议，并可联合有关主管部门根据漏洞影响范围及发展情况制定下一步处置方案和解决措施；
e) 提供有效且便利的途径，在安全的通道下让用户获取补丁、升级版本和临时处置建议；
f) 宜调查漏洞更深层的原因，以及确定自身其他产品或服务是否有同样或者类似的漏洞；
g) 宜实时关注 App 使用的框架、组件等相关版本安全动态。

6.2.2.4 安全漏洞的跟踪

在漏洞发布后开发者持续对漏洞进行跟踪监测，收集用户反馈信息，确保应用稳定运行，并视情况对漏洞修复或防范措施做进一步改进，确认需改进时漏洞管理流程再次进入漏洞处置阶段。

附 录 A
（资料性）
App面临的安全风险

表A.1给出了恶意程序的分类及风险描述。

表A.1 恶意程序的分类及风险描述

恶意程序分类（包括但不限于以下分类）	风险描述
恶意扣费	在用户不知情或未授权的情况下，通过隐蔽执行、欺骗用户点击等手段，订购各类收费业务或使用移动智能终端支付，导致用户经济损失。
信息窃取	在用户不知情或未授权的情况下，获取用户个人信息、工作信息或其他非公开信息。
远程控制	在用户不知情或未授权的情况下，接受远程控制端指令并进行相关操作。
恶意传播	在用户不知情或未授权的情况下，自动通过复制、感染、投递、下载等方式扩散恶意程序自身、自身的衍生物或其他恶意程序。
资费消耗	在用户不知情或未授权的情况下，自动通过拨打电话、发送短信、彩信、邮件、频繁连接网络等方式消耗用户资费。
系统破坏	在用户不知情或未授权的情况下，通过感染、劫持、篡改、删除、终止进程等手段使移动智能终端或其他非恶意程序部分或全部功能、用户文件等无法正常使用，干扰、破坏、阻断移动通信网络、网络服务或其他合法业务的正常运行。
诱骗欺诈	在用户不知情或未授权的情况下，通过伪造、篡改、劫持短信、彩信、邮件、通信录、通话记录、收藏夹、桌面等方式诱骗用户，达到不正当目的。
流氓行为	在用户不知情或未授权的情况下，执行对系统没有直接损害，也不对用户个人信息、资费造成侵害的其他恶意行为。

表A.2给出了侵害用户权益的分类及风险描述。

表A.2 侵害用户权益的分类及风险描述

侵害用户权益分类（包括但不限于以下分类）	风险描述
私自收集个人信息	App未明确告知收集使用个人信息的目的、方式和范围并获得用户同意前，收集用户个人信息。
超范围收集个人信息	App收集个人信息，非服务所必需或无合理应用场景，超范围或超频次收集个人信息，如通讯录、位置、身份证、人脸等。
私自共享给第三方	App未经用户同意与其他应用共享、使用用户个人信息，如设备识别信息、商品浏览记录、搜索使用习惯、常用软件应用列表等。
强制用户使用定向推送功能	App未向用户告知，或未以显著方式标示，将收集到的用户搜索、浏览记录、使用习惯等个人信息，用于定向推送或精准营销，且未提供关闭该功能的选项。
不给权限不让用	App安装和运行时，向用户索取与当前服务场景无关的权限，用户拒绝授权后，应用退出或关闭。

表 A.2（续）

频繁申请权限	App在用户明确拒绝权限申请后，频繁申请开启通讯录、定位、短信、录音、相机等与当前服务场景无关的权限，骚扰用户。
过度索取权限	App在用户未使用相关功能或服务时，提前申请开启通讯录、定位、短信、录音、相机等权限，或超出其业务功能或服务外，申请通讯录、定位、短信、录音、相机等权限。
账号注销难	App未向用户提供账号注销服务，或为注销服务设置不合理的障碍。
设置障碍、频繁骚扰用户	App强制、频繁、过度索取权限，App频繁自启动和关联启动。
欺骗误导用户	欺骗误导用户下载App，欺骗误导用户提供个人信息。
移动应用分发平台责任落实不到位	移动应用分发平台上的App信息明示不到位，应用分发平台管理责任落实不到位。
其他风险行为	包括但不限于：生成、复制、发布的内容中存在违法信息或不良信息的行为，使用无支付牌照的支付平台、支付渠道、参与非法洗钱过程的行为，使用诱导、欺诈、迷惑、操纵等方式对用户实施资金诈骗的行为，或通过加密数据方式向用户勒索钱财的行为等。

表A.3给出了安全漏洞的分类及风险描述。

表A.3 安全漏洞的分类及风险描述表

安全漏洞分类（包括但不限于以下分类）	风险描述
资源管理错误	因对系统资源（如内存、文件、CPU使用率等）的错误管理导致的漏洞。
输入验证错误	因对输入的数据缺少正确的验证而产生的漏洞，包括但不限于：缓冲区错误、注入、路径遍历、后置链接、跨站请求伪造等。
数字错误	因未正确计算或转换所产生数字，导致的整数溢出、符号错误等漏洞。
竞争条件问题	因为在并发运行环境中，一段并发代码需要互斥地访问共享资源时，因另一段代码在同一个时间窗口可以并发修改共享资源而导致的安全问题。
处理逻辑错误	在设计实现过程中，因处理逻辑实现问题或分支覆盖不全面等原因造成。
加密问题	未正确使用相关密码算法，导致的内容未正确加密、弱加密、明文存储敏感信息等问题。
信任管理问题	缺乏有效的信任管理机制，导致受影响组件存在可被攻击者利用的默认密码或者硬编码密码、硬编码证书等问题。
权限许可和访问控制问题	因缺乏有效的权限许可和访问措施而导致的安全问题。
数据转换问题	程序处理上下文因对数据类型、编码、格式、含义等理解不一致导致的安全问题。
未声明功能	通过测试接口、调试接口等可执行非授权功能导致的安全问题。例如，若测试命令或调试命令在使用阶段仍可用，则可被攻击者用于显示存储器内容或执行其他功能。
配置错误	App或组件在使用过程中因配置文件、配置参数或因默认不安全的配置状态而产生的漏洞。
信息泄露	在运行过程中，因配置等错误导致的受影响组件信息被非授权获取的漏洞，包括但不限于：日志信息泄露、调试信息泄露、侧信道信息泄露。

表A.3（续）

故障注入	通过改变运行环境出发，可能导致代码、系统数据或执行过程发生错误的安全问题。
其他	暂时无法将漏洞归入上述任何类别，或者没有足够充分的信息对其进行分类，漏洞细节未指明。

附 录 B
（资料性）
App 安全风险应对体系

为了降低各阶段可能产生的风险，表B.1给出了App安全风险应对体系。

表B.1 App 安全风险应对体系

安全保证模块	风险问题	风险产生阶段
安全开发	系统破坏	代码实现、输入输出、异常处理、开发环境、运行环境、第三方SDK或组件管理
	信息窃取	代码实现、输入输出、异常处理、端口安全、会话安全、传输过程安全、身份鉴别、口令安全、权限管理、日志记录与保护、数据真实性、数据完整性、数据保密性、不可否认性、密码支持、运行环境、第三方SDK或组件检测、第三方SDK或组件调用、收集与使用、删除与保护
	远程控制	代码实现、输入输出、端口安全、会话安全、身份鉴别、口令安全、权限管理、开发环境、第三方SDK或组件调用、发布安全
	诱骗欺诈	会话安全、传输过程安全、身份鉴别、权限管理
	恶意扣费	会话安全、传输过程安全
	资费消耗	会话安全、传输过程安全
	恶意传播	开发环境、测试环境、第三方SDK或组件管理、第三方SDK或组件检测
	安全漏洞	运行环境、第三方SDK或组件管理、第三方SDK或组件检测、需求分析及评审、安全质量保证
	流氓行为	发布安全
生命周期管理	恶意程序（详见表A.1）	需求分析及评审、安全质量保证、方案设计、方案评审、代码管理、变更控制、安全测试、安全验证、安全交付、App管理、整改监督、安装检测、运行状态检测、安全维护、安全更新、终止运营、安全管理制度、人员管理、文档管理、业务数据管理、
	侵害用户权益（详见表A.2）	需求分析及评审、安全质量保证、方案设计、方案评审、代码管理、变更控制、安全测试、安全验证、安全交付、App管理、整改监督、安装检测、运行状态检测、安全维护、安全更新、终止运营、安全管理制度、人员管理、文档管理、业务数据管理、个人信息管理、侵害用户权益风险监测处置过程
	安全漏洞（详见表A.3）	需求分析及评审、安全质量保证、方案设计、方案评审、代码管理、变更控制、安全测试、安全验证、安全交付、App管理、整改监督、安装检测、运行状态检测、安全维护、安全更新、终止运营、安全管理制度、文档管理、漏洞管理、业务数据管理、安全漏洞风险监测处置
	安全检测被绕过	上架审核

附 录 C
（资料性）
安全开发

C.1 程序安全

C.1.1 编码安全

C.1.1.1 代码实现

App在代码实现时除了参考GB/T 38674—2020 6.1、6.2、6.3、6.5，还宜参考以下方面的内容：

a) 随机生成代码中的初始化矢量参数；
b) 仅将可信数据解析为命令或查询语句；
c) 不将密钥或敏感信息直接编写在代码中；
d) 当使用安全传输协议进行通信时，执行验证策略校验服务端证书的合法性；
e) 当调用 SDK 时，验证 SDK 签名的有效性；
f) 当动态加载代码时，验证代码来源的安全性，设置被加载代码的权限不超过 App 本身的权限；
g) 当导出代码或数据时，启用访问控制机制；
h) 在发布代码前，删除代码中出现的调试信息、测试数据和含有敏感信息的代码注释等。

C.1.1.2 输入输出

App宜采用以下方法：

a) 过滤或标准化处理输入数据；
b) 验证输入数据的有效性，如数据类型、格式、长度、特殊字符和大小写等；
c) 当进行重定向操作时，验证输入数据的安全性，如检查是否存在恶意程序、不可信站点或任意重定向等；
d) 当进行重要业务操作时，验证输入数据的真实性和完整性，如采用数字签名方法等；
e) 过滤或正确转化向页面输出的数据。

C.1.1.3 异常处理

App在处理异常时宜参考以下内容：

a) 处理子线程的异常时不对主线程的运行产生影响；
b) 告知用户异常，并使用通用的错误消息或定制的错误页面，不展示详细错误提示或敏感信息；
c) 兼顾容错性，不因可预知的错误操作影响正常运行；
d) 服务器收集异常信息时，保障数据存储和传输的安全。

C.1.2 通信安全

C.1.2.1 端口安全

App在设置端口时宜参考以下内容：

a) 关闭不必要得服务器远程端口，包括但不限于调试端口；
b) 不将多个套接字绑定到同一端口；
c) 当使用服务器远程端口时，认证端口连接对象的身份，并启用鉴权机制。

C.1.2.2 会话安全

App在会话安全方面宜采用以下设置方法：

a) 设置唯一、随机、无法识别的会话标识符；
b) 当进行重要业务操作时，如移动支付和身份认证等，使用随机令牌或参数；
c) 限制最大并发会话连接数；
d) 宜设置网络传输流量上限值；
e) 当重新登录时，关闭之前的会话，创建新会话。

C.1.2.3 传输过程安全

App在传输过程中宜采用以下方法：

a) 使用时间戳、随机数等数据保护传输内容；
b) 宜采用数字证书签名验证数据可靠性和有效性；
c) 当采用安全传输协议进行通信时，使用有效的安全协议、安全配置和验证策略。

C.1.3 访问控制

C.1.3.1 身份鉴别

App在执行身份鉴别时宜参考以下方面：

a) 对每个用户身份进行唯一标识，并分配相应的访问权限；
b) 宜根据场景（如涉及交易、支付等）安全需求选择多因素鉴别方式，如口令鉴别、基于令牌的动态口令鉴别、指纹或虹膜等生物特征鉴别、数字证书鉴别或图形鉴别等；
c) 当使用数字证书时，检查证书的状态和证书持有者的有效性；
d) 当身份鉴别失败时，启用身份鉴别失败次数限制或增加人机验证；
e) 当执行修改鉴别信息、转账或支付等敏感操作时，再次对用户身份进行鉴别。

C.1.3.2 口令安全

App在设置口令时宜参考以下方面：

a) 口令的复杂度根据场景需求选择符合安全策略；
b) 不以明文的方式显示、存储、传输口令；
c) 不默认缓存、填充口令信息；
d) 当采用动态口令时，设置口令的最长有效期限和生成口令的最短时间间隔；
e) 当重置口令时，对用户身份再次进行鉴别，宜随机生成重置口令问题，并对发送重置口令链接的邮箱或电话等信息进行脱敏显示。

C.1.3.3 权限管理

App在权限的管理上宜采用以下方法：

a) 动态申请业务所需权限；
b) 仅申请业务所需的最小权限（或最低的安全许可），不申请与业务功能无关的其他权限；
c) 告知用户权限申请目的，且目的描述易于理解，而不欺诈、诱骗或误导用户；
d) 当权限申请目的或使用场景发生变化时，再次告知用户；
e) 不以默认、捆绑、停止安装使用等手段变相强迫用户授予权限；

f) 当用户拒绝授权或撤回授予的非必要权限时，不强制退出、关闭App，不影响与该权限无关的其他业务功能使用；
g) 当使用权限时，仅使用与申请声明目的、方式和范围一致的权限；
h) 当使用权限访问个人信息时，仅访问满足业务功能所需的最少信息；
i) 不采用非法手段绕过系统的权限管控机制，如绕过用户授权对用户行为的监听、录音、录像和启动套接字等；
j) 当需要使用系统不开放的敏感权限或能力时，提前向系统厂商申请授权；
k) 验证所授予权限与重要数据的访问或操作权限的一致性，并记录权限与操作过程，对访问或操作重要数据的状态进行完整性保护。

C.1.4 日志记录与保护

App在日志的记录与保护上宜采用以下方法：
a) 记录用户操作的日志信息，如操作的主体、客体、访问行为和时间戳等；
b) 记录安全事件的日志信息，如安全事件发生的日期、时间、去标识化的主体信息、事件类型描述、结果（成功或失败）和关联的进程信息等；
c) 不记录未做去标识化或匿名化的个人信息；
d) 仅授权用户可访问日志文件。

C.1.5 数据保护与密码

C.1.5.1 数据真实性

App宜采用密码技术保护数据的真实性：
a) 在通信前基于密码技术对通信的双方进行验证或认证；
b) 采用口令、密码技术、生物技术等两种或两种以上组合的鉴别技术对用户进行身份鉴别，且其中一种鉴别技术至少应使用密码技术来实现。

C.1.5.2 数据完整性

App宜采用密码技术保护以下数据的完整性：
a) 重要的存储数据，如鉴别数据、重要业务数据、重要审计数据、重要配置数据、重要视频数据和个人敏感信息等，并检测数据完整性被破坏的情况；
b) 重要的传输数据，并检测数据被篡改、删除和插入等情况；
c) 执行回退操作序列中的回退数据。

C.1.5.3 数据保密性

App宜采用密码技术保护以下数据的保密性：
a) 存储数据加密，包括但不限于有版权保护的音视频服务数据、移动支付数据、网上购物服务数据、鉴别数据、重要业务数据和个人敏感信息等，并确保无访问权限的用户无法获取该数据；
b) 传输数据加密，并确认在传输过程中窃取或泄漏重要数据的情况；
c) 内存中的个人敏感信息和重要数据，并检测该数据在内存被释放或重新分配前得以清除。

C.1.5.4 不可否认性

在可能涉及法律责任认定需要对行为进行取证时，App宜采用密码技术提供数据原发证据和数据接收证据，实现数据原发行为的不可否认性和数据接收行为的不可否认性。

C.1.5.5 密码支持

App采用密码技术时宜参考以下方面：

a) 仅使用符合法律、法规规定和密码相关国家标准、行业标准有关要求的密码算法、密码技术、密码产品和密码服务；

b) 建立与业务场景相适应的密钥管理机制，并由随机或协商等方式生成密钥。

C.2 安全保障

C.2.1 环境安全

C.2.1.1 开发环境

App在开发环境方面宜采用以下方法：

a) 通过官方渠道下载安全稳定的、已安装补丁的开发工具、编译器和框架等；

b) 使用编译器的安全编译选项，关闭编译器中不必要的编译功能；

c) 不使用有已知安全风险的依赖库；

d) 使用具有安全功能的、版本稳定的 API，不使用废弃的或未公开的 API，API 宜兼容最新移动智能终端操作系统；

e) 在独立的模拟环境中编写、调试和完成代码，仅授权的开发和测试团队可访问该模拟环境；

f) 保护开发环境中的重要配置数据。

C.2.1.2 运行环境

App在运行环境方面宜采用以下方法：

a) 不使用客户端的超级用户权限或 App 间的守护进程；

b) 宜使用混淆、签名、加固等措施保护客户端的源代码；

c) 宜使用模拟器框架等虚拟化方法保护客户端不被模拟运行；

d) 使用安全稳定的、已安装补丁的服务器系统组件，不使用存在已知漏洞的服务器系统组件；

e) 保护运行环境中的重要配置数据，如缓存、虚拟化的系统隔离环境中的重要数据。

C.2.1.3 测试环境

App在测试环境方面宜采用以下方法：

a) 使用安全的开发环境或仿真环境；

b) 使用没有已知安全风险的测试工具。

C.2.2 第三方SDK或组件安全

C.2.2.1 第三方 SDK 或组件管理

App对第三方SDK或组件宜采用以下管理方法：

a) 进行安全风险管理，识别和评估可能引入的安全风险，不引入不可接受的风险；

b) 进行权限管理，如统一通过宿主 App 声明权限、不热更新 SDK 等；

c) 修复发现的安全问题，并更新代码。

C.2.2.2 第三方 SDK 或组件检测

App检测第三方SDK或组件时除了参考SDK自身安全的其他相关标准，还宜参考以下方面的内容：

a) 验证代码的完整性和许可证的有效性；
b) 进行代码安全性检测，不引入恶意程序、安全漏洞或其他潜在安全风险；
c) 进行行为安全性检测，不引入后台自启动或关联启动、超出实现业务所需功能范围的其他代码；
d) 检查权限申请与声明的一致性，并符合 C.1.3.3 的权限管理；
e) 检查个人信息收集使用与声明的一致性，并符合 C.2.3.2 的收集与使用；
f) 采用漏洞检测工具、软件分析工具等定期进行安全检测，或通过有资质的第三方安全检测机构的检测认证。

C.2.2.3 第三方 SDK 或组件调用

App宜采用以下方法：

a) 使用代码混淆、加壳或加密等反编译技术；
b) 启用接口鉴权机制，对不同宿主 App 调用接口的上下文环境进行隔离；
c) 使用安全信道、数字证书双向校验或证书绑定等方式传输数据；
d) 当停用第三方 SDK 或组件时，及时从宿主 App 中移除代码。

C.2.3 发布安全

为实现代码的安全稳定，App发布时宜采用以下方法：

a) 在安装包中写入不能被篡改的开发者信息，并控制源代码版本；
b) 对安装包的 Hash 值进行校验，确保发布的安装包来源于受控的版本；
c) 关闭调试接口，不以测试模式、任意备份模式和任意调试模式等状态发布；
d) 删除调试日志，不在安装包中保留代码注释；
e) 当用户没有授权时，不修改已有的安全配置数据。

C.2.3.1 个人信息安全

C.2.3.2 收集与使用

除了符合GB/T 35273—2020的基本原则，App还宜参考以下方面：

a) 在隐私政策中公开收集使用个人信息规则，并在首次运行时通过弹窗等明显方式提示用户阅读；
b) 告知用户收集、使用个人信息的目的、方式和范围等，并在获得用户授权同意后才收集个人信息；
c) 收集的个人信息与声明的目的、方式和范围等保持一致；
d) 当个人信息的使用目的、范围或场景发生变化时，再次征得用户明示同意；
e) 不骚扰用户频繁征求用户的授权；
f) 不诱导或强迫用户授权同意，如通过捆绑产品或服务各项业务功能的方式诱导用户，或通过改善服务质量、提升使用体验、研发新产品、增强安全性或暂停其他业务功能等方式强迫用户；
g) 不超范围收集个人信息，如非服务所必需或无合理应用场景的个人信息；
h) 不通过隐蔽方式收集，如个人信息的字段、类型、收集频率、回传服务端场景信息和内容等。

C.2.3.3 删除与保护

为实现个人信息的删除与保护， App宜采用以下方法：

a） 提供用户查询、更正、删除其个人信息的方式；

b） 提供用户撤回收集、使用其个人信息的授权同意的方式；

c） 提供公布个人信息安全投诉、举报的方式；

d） 当用户注销账户时，删除或匿名化处理其个人信息；

e） 启用访问控制机制处理个人信息。

参 考 文 献

[1] GB/T 8566—2007 信息技术 软件生存周期过程

[2] GB/T 15532—2008 计算机软件测试规范

[3] GB/T 20269—2006 信息安全技术 信息系统安全管理要求

[4] GB/T 20271—2006 信息安全技术 信息系统通用安全技术要求

[5] GB/T 20984—2007 信息安全技术 信息安全风险评估规范

[6] GB/T 22032—2008 系统工程 系统生存周期过程

[7] GB/T 22239—2019 信息安全技术 网络安全等级保护基本要求

[8] GB/T 28172—2011 嵌入式软件质量保证要求

[9] GB/T 28452—2012 信息安全技术 应用软件系统通用安全技术要求

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

[11] GB/T 32421—2015 软件工程 软件评审与审核

[12] GB/T 33132—2016 信息安全技术 信息安全风险处理实施指南

[13] GB/T 34975—2017 信息安全技术 移动智能终端应用软件安全技术要求和测试评价方法

[14] GB/T 34978—2017 信息安全技术 移动智能终端个人信息保护技术要求

[15] GB/T 37964—2019 信息安全技术 个人信息去标识化指南

[16] GB/T 38674—2020 信息安全技术 应用软件安全编程指南

[17] GB/T 39786—2021 信息安全技术 信息系统密码应用基本要求

[18] YD/T 2439—2012 移动互联网恶意程序描述格式

[19] GB/T 39335—2020 信息安全技术 个人信息安全影响评估指南

[20] GB/T 39412—2020 信息安全技术 代码安全审计规范

[21] GB/T 39720—2020 信息安全技术 移动智能终端安全技术要求及测试评价方法

[22] GB/T 28458—2020 信息安全技术 网络安全漏洞标识与描述规范

[23] GB/T 35273—2020 信息安全技术 个人信息安全规范

[24] YD/T XXXXX—XXXX 移动互联网应用程序（APP）用户权益保护测评规范

[25] TC260-PG-20191A 网络安全实践指南—移动互联网应用基本业务功能必要信息规范

[26] App违法违规收集使用个人信息行为认定方法（国信办秘字〔2019〕191号），2019年11月28日

[27] 中华人民共和国网络安全法（中华人民共和国主席令第53号），2017年6月1日

[28] 移动智能终端应用软件预置和分发管理暂行规定（工信部信管〔2016〕407号），2016年12月16日

[29] 关于开展纵深推进APP侵害用户权益专项整治行动的通知（工信部信管函〔2020〕164号），2020年7月22日