ICS 33.050
CCS M 30

团　体　标　准

T/TAF 162—2023

移动应用安全平行切面技术指南

Technical guidelines for aspect-oriented security of mobile applications

2023-04-26 发布　　2023-04-26 实施

电信终端产业协会　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。

本文件由电信终端产业协会提出并归口。

本文件起草单位：蚂蚁科技集团股份有限公司、中国信息通信研究院、联想(北京)有限公司、北京快手科技有限公司、维沃移动通信有限公司、北京抖音信息服务有限公司、上海兆言网络科技有限公司。

本文件主要起草人：白晓媛、郑旻、韦韬、刘陶、傅山、宋玉成、靳宇星、李婷婷、彭晋、李汝鑫、落红卫、王昕、贾科、李昳婧、杜蕾、钱雷、刘微。

移动应用安全平行切面技术指南

1 范围

本文件规定了移动应用安全平行切面架构，并从性能、稳定性、安全性等方面给出移动应用安全平行切面的部署指南。

本文件适用于移动应用开发者设计、开发、测试移动应用时做参考，也适用于第三方评估机构对移动应用安全测评时做参考。

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

3 术语和定义

GB/T 25069界定的以及下列术语和定义适用于本文件。

3.1

安全平行切面 aspect-oriented security

通过在关键 API 函数和业务逻辑层间插入安全管控函数，在运行时对目标函数进行监测和管控，进而实现安全风险感知和阻断的技术体系。

4 缩略语

下列缩略语适用于本文件。

AOP：面向切面编程（aspect oriented programming）

AOS：安全平行切面（aspect-oriented security）

API：应用程序接口（application programming interface）

App：移动互联网应用程序（mobile internet application）

IAST：交互式应用安全测试（interactive application security testing）

RASP：运行时应用自防护（runtime application self-protection）

SDK：软件开发工具包（software development kit）

5 概述

移动应用特别是平台型App，存在大量第三方SDK、小程序等第三方代码。由于缺少运行时监测技术，无法对实际调用行为进行观测，存在盗取个人信息、推送恶意广告等风险，严重影响App的安全性。

传统方式下，App安全风险分析主要针对缺少安全管控或者管控存在疏漏的输入与输出，但是存在一些困难：

a) 难以枚举应用的输入与输出。静态分析可以枚举出一部分，但是覆盖率有限且存在一定程度误报；

b) 复现困难。App的某些行为，仅在特殊场景下才会触发，安全分析人员难以了解每个业务的细节，这些仅在特殊场景下触发的行为难以发现和评估。

安全平行切面（AOS）将面向切面编程（AOP）思想应用到安全体系建设中，通过端—管—云各层次切面使安全管控与业务逻辑相互融合且解耦，并通过标准化接口为业务提供运行时监测与风险干预能力。

注：AOP技术简介参见附录A。

面向App的AOS可以提供App运行时数据监测、行为刻画和干预能力，实现针对运行时的威胁发现和恶意收集个人信息行为的监测和防护。其关键技术点如下：

a) 切点植入：选取业务关键逻辑点植入安全防护切点，切点植入支持多种方式，如流量层接入、动态接入和静态框架接入等。

b) App运行时数据监测：安全平行切面深入App内部，不需要经过序列化或反序列化等转换过程即可直接获取到App运行时的数据（App内部函数的调用链路、关键API的调用参数等），提升安全治理效果。

c) 风险干预：能够实现对异常行为和安全风险的实时感知，通过切面对函数调用进行拦截管控，从而对有风险的隐私泄露或漏洞攻击链路进行管控。

6 移动应用安全平行切面架构

6.1 概述

移动应用安全平行切面主要由移动端切面组件和云端切面管控平台等组成，如图1所示。移动应用安全平行切面所提供的App运行时数据监测和风险干预能力，可以实现不同的安全业务能力，比如隐私权限管控、RASP和IAST等。

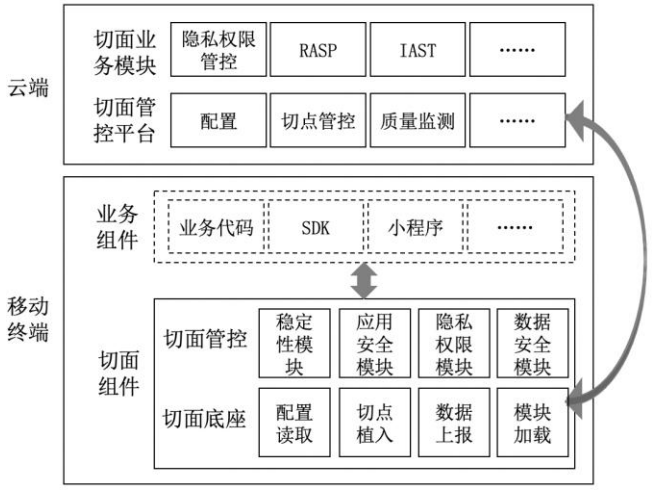

图1 移动应用安全平行切面架构

6.2 切面组件

切面组件分为切面底座和切面管控两部分。其中切面底座主要负责配置读取、切点植入、数据上报和模块加载等功能，切面管控负责具体的切点处理逻辑。不同的切面管控模型负责不同的安全业务场景。在App运行期间，切面管控模块通过调用栈、函数参数、返回值和上下文等信息可以获取到每一条调研链路的详情，针对不同的安全业务场景比如隐私权限API调用、数据和网络请求等做出对应的分析判断，并通过切面底座完成数据上报。

切面底座功能包括但不限于：

a) 配置读取：读取云端切面管控平台下发的切面配置信息；
b) 切点植入：通过预编译、运行时动态代理等方式植入切点；
c) 数据上报：将切面管控模块读取到的数据上报云端切面管控平台；
d) 模块加载：加载切面管控中的相应模块。

切面管控功能包括但不限于：

a) 稳定性模块：通过在业务组件中设置安全平行切面，监测线程相关的API，阻止对业务组件稳定性造成影响的线程的运行；
b) 应用安全模块：将业务组件（如小程序）的关键API函数（如文件访问等）中植入安全管控函数，加入安全监测和拦截的功能，从而实现移动端运行时自防护；
c) 隐私权限模块：通过在涉及到用户隐私数据访问的业务组件中设置安全平行切面，对隐私权限相关API调用进行监测和管控；
d) 数据安全模块：通过在不同业务组件（如App中的小程序）中设置安全平行切面，返回不同模块调用云端API的信息给云端，帮助云端实现对不同业务组件的数据访问控制。

6.3 切面管控平台

切面管控平台在获取到上报的数据后，通过大数据分析和算法模型刻画出业务的行为特征并筛选出异常行为，必要时可下发对应的配置到移动端切面组件，对有风险的隐私泄露或漏洞攻击等风险进行阻断。功能包括但不限于：

a) 配置：针对移动端的异常行为，生成相应的移动端切面配置信息并下发到移动端；
b) 切点管控：根据移动端异常行为或安全漏洞信息，生成切点列表并下发到移动端；
c) 质量监测：通过对加载了切面的业务组件的相关信息的统计和分析（比如时延、运行异常等信息），对切面的运行进行监测，避免切面的运行影响到业务的正常运行。

6.4 切面业务模块

切面业务即基于安全平行切面实现的App安全业务能力，不同的切面业务之间相互独立，互不干扰。功能包括但不限于：

a) 隐私权限管控：对隐私相关的API函数的信息进行规则匹配和管控，下发隐私权限管控指令；
b) RASP：对漏洞相关的API函数的信息进行规则匹配和管控，下发安全防护指令；
c) IAST：交互式应用安全测试，主要应用于测试环境，下发测试数据，用于发现可能导致App崩溃的漏洞。

7 部署指南

7.1 性能保障措施

在安全平行切面实践中，主要从以下方面控制和优化性能开销：

a) 宜避免引入超高频次调用的切点，通过关注和实测切点被调用的频次，来选择合适粒度的切点；
b) 宜建立安全平行切面组件分级分类管理，区分不同切面组件的能力边界，防止切面组件被滥用；
c) 宜优先使用经过实践验证过的轻量级切点植入机制或框架，将植入动作所带来的开销降到最低；
d) 宜权衡和控制切点出植入代码的复杂度和收益，不过度追求内视和安全能力最大化，避免引入非常复杂的安全逻辑。

7.2 稳定性保障措施

在安全平行切面实践中，主要从以下方面保障切面和系统的运行稳定：

a) 宜在研发阶段加强质量管理，建立研发标准规范，收敛研发过程中出现的稳定性风险；
b) 安全平行切面进行发布变更或策略变更时，宜引入相关方交叉评审，确保充分评估风险，控制影响范围；
c) 宜具备完善的变更防御机制、监控告警和保障机制，建立运营保障规范，安全策略和变更过程要符合可监控、可回滚的稳定性要求；
d) 宜进行切面使用资源的限制与隔离，避免过度使用控制资源，避免切点隐蔽死锁问题，避免设计过度复杂的安全逻辑等；
e) 宜通过链路压测、故障演练、预案演练等手段模拟真实流量压力和故障，在模拟故障产生后，全面评估各项告警的可靠性、应急处置动作的及时性、故障预案执行的有效性，从而检验安全平行切面的稳定性和有效性；
f) 在运行时，宜通过监控切面指标（如切面注入点耗时、切面抛出异常数、切面拦截量等）和业务进程自身的系统指标（CPU 占用、内存消耗、服务耗时等），进行多层次的综合判定安全平行切面运行稳定性；
g) 宜建立统一的故障处置应急机制，对安全平行切面故障进行快速处置，保障业务可用性和稳定性。比如根据业务运行告警和异常情况，关闭切点处的安全逻辑，保障业务运行的稳定性。
a) 宜考虑碎片化场景需求，适配不同版本的基础环境，为上层业务提供相对统一的安全平行切面机制，为业务系统的快速迭代提供高效稳定的安全保障。

7.3 安全保障措施

在安全平行切面实践中，主要从以下方面进行安全保障

a) 宜对切面组件和安全策略进行签名验证，防止被篡改和滥用；
b) 宜对切面组件的安装升级和安全策略下发等操作进行身份鉴别和权限管控，落实到人，降低被攻击者冒用的风险；
c) 宜对切面获取到的隐私API调用相关信息的传输和存储进行加密保护，防止敏感数据泄露；
d) 宜对移动终端和云端之间的切面数据上报和指令下发等进行传输加密；
e) 宜防止切面管控被攻击者绕过；
f) 宜采用代码混淆等技术，加大攻击者的攻击难度，实现对切面组件的保护；
g) 宜对切面管控平台及其所有操作进行严格的身份鉴别和权限管控，权限检查内置与安全平行切面中，并可实现针对切面业务模块的细化权限；
h) 宜对所有云端和移动端的配置变更、状态变更进行日志审计。

附 录 A
（资料性）
AOP技术简介

AOP是一种编程范式，主要基于OOP（面向对象编程）进行延续。OOP从纵向上区分出一个个的类，让开发者实现纵向的业务逻辑处理，但是OOP并不适合用于定义横向业务逻辑的关系，特别是实现涉及大量类的横切（cross cutting）功能逻辑时，会导致程序中出现大量重复代码，复用性极差，如最常用的日志、安全以及事务功能等，它们都可能是横向的分布在不通的业务层级中，但是又和具体的核心业务无直接关系，诸如这样类型的代码，在程序中被称作横切。

AOP就是将此类与核心业务无关的，但又影响着多个类的公共行为抽取、封装到一个可重用公共模块中，通过预编译、运行时动态代理、注入等方式，在不修改源代码的情况下给程序的正常业务逻辑中动态添加公共模块中定义的功能。从而实现代码复用和解耦的目的。

在AOP中，切面是通知（advice）和切点（point cut）的集合。“通知”描述了切面要实现的功能和执行此功能的时机，比如在函数执行完成之后调用日志功能，即“通知”定义了“什么”和“何时”。“切点”主要定义“何处”，即在何处调用横切功能，比如名字前面是select开头的函数才调用日志功能。

参 考 文 献

[1] 安全平行切面白皮书， http://www.itstec.org.cn/aspect_oriented_security_white_paper.pdf

电信终端产业协会团体标准

移动应用安全平行切面技术指南

T/TAF 162—2023

*

电信终端产业协会印发
地址：北京市西城区新街口外大街 28 号
电话：010-82052809
电子版发行网址：www.taf.org.cn