云计算开源产业联盟

OpenSource Cloud Alliance for industRy,OSCAR

研发运营安全白皮书

（2020年）

云计算开源产业联盟

OpenSource Cloud Alliance for industry，OSCAR

2020年7月

版权声明

前　言

近年来，安全事件频发，究其原因，软件应用服务自身存在代码安全漏洞，被黑客利用攻击是导致安全事件发生的关键因素之一。随着信息化的发展，软件应用服务正在潜移默化的改变着生活的各个方面，渗透到各个行业和领域，其自身安全问题也愈发成为业界关注的焦点。传统研发运营模式之中，安全介入通常是在应用系统构建完成或功能模块搭建完成之后，位置相对滞后，无法完全覆盖研发阶段的安全问题。在此背景下，搭建整体的研发运营安全体系，强调安全左移，覆盖软件应用服务全生命周期安全，构建可信理念是至关重要的。

本白皮书首先对于研发运营安全进行了概述，梳理了全球研发运营安全现状，随后对于信通院牵头搭建的研发运营安全体系进行了说明，归纳了研发运营安全所涉及的关键技术。最后，结合当前现状总结了研发运营安全未来的发展趋势，并分享了企业组织研发运营安全优秀实践案例以供参考。

参与编写单位

中国信息通信研究院、华为技术有限公司、深圳市腾讯计算机系统有限公司、阿里云计算有限公司、浪潮云信息技术股份公司、京东云计算（北京）有限公司、北京金山云网络技术有限公司、深圳华大生命科学研究院、奇安信科技集团股份有限公司、杭州默安科技有限公司、新思科技（上海）有限公司

主要撰稿人

吴江伟、栗蔚、郭雪、耿涛、康雪婷、徐毅、章可镌、沈栋、郭铁涛、张祖优、马松松、黄超、伍振亮、祁景昭、朱勇、贺进、宋文娣、张娜、蔡国瑜、张鹏程、张玉良、董国伟、周继玲、杨国梁、肖率武、薛植元

目 录

图 目 录

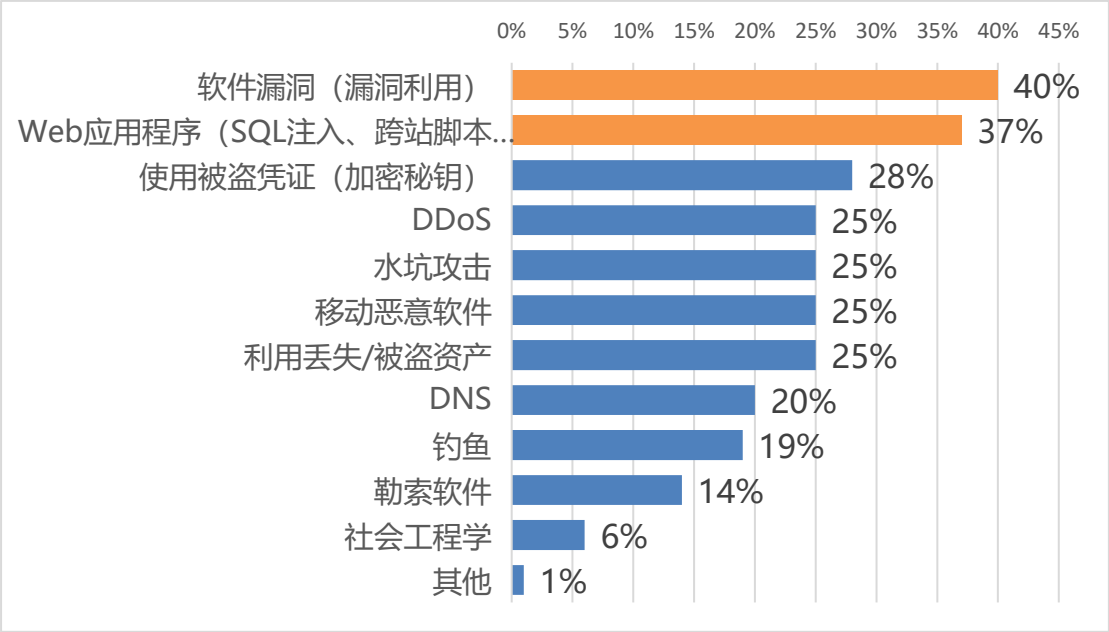
表 目 录

一、 研发运营安全概述

（一）研发层面安全影响深远，安全左移势在必行

随着信息化的发展，软件应用服务正在潜移默化的改变着生活的各个方面，渗透到各个行业和领域，软件应用服务的自身安全问题也愈发成为业界关注的焦点。

全球安全事件频发，代码程序漏洞是关键诱因之一。2017 年，美国最大的征信机构之一 Equifax 因未能及时修补已知的安全漏洞发生一起涉及 1.48 亿用户的数据安全、隐私泄露事件，影响几乎一半的美国人口；国内电商因优惠券漏洞被恶意牟利，酒店、求职等网站也曾发生数据安全事件，泄露百万级、亿级用户隐私数据。究其原因，软件应用服务自身安全漏洞被黑客利用攻击是数据安全事件层出不穷关键因素之一。根据 Verizon 2019 年的研究报告《Data Breach Investigations Report》，在总计核实的 2013 次数据泄露安全事件中，超过 30%与 Web 应用程序相关，Web 应用程序威胁漏洞具体指程序中的代码安全漏洞以及权限设置机制等。Forrester 2019 年发布的调查报告《Forrester Analytics Global Business Technographics Security Survey，2019》中显示，在 283 家全球企业已经确认的外部攻击中，针对软件漏洞以及 Web 应用程序是位于前两位的，分别占比达到了 40%与 37%，具体数据见图 1，其中软件漏洞主要指对于安全漏洞的利用攻击，攻击 Web 应用程序主要指基于程序的 SQL 注入、跨站脚本攻击等。



数据来源：Forrester

图 1 Forrester 外部攻击对象统计数据

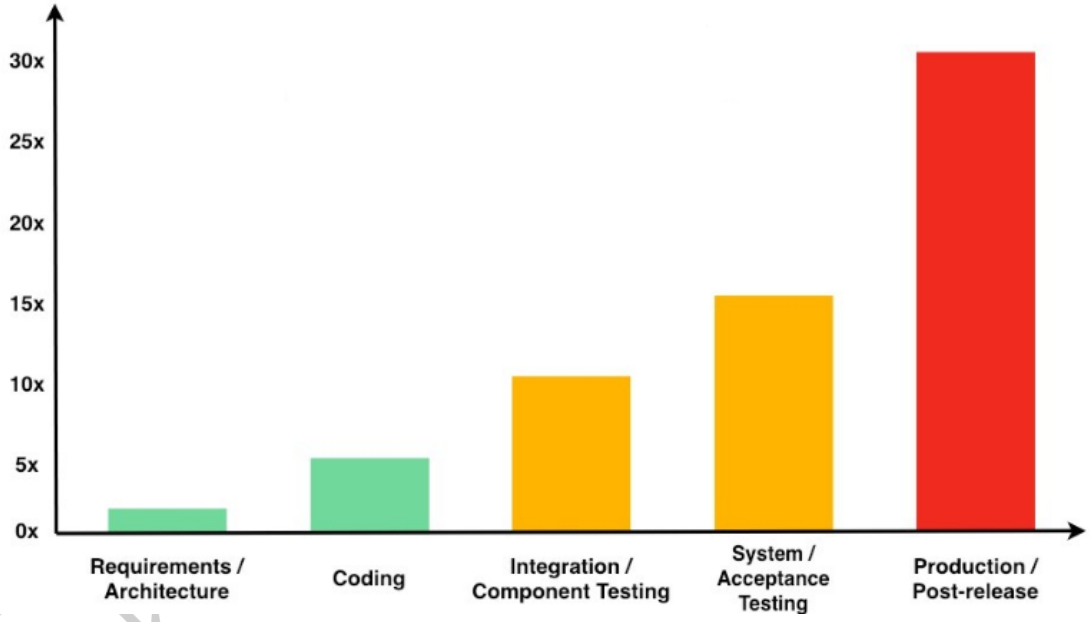
根据咨询公司 Gartner 统计数据显示，超过 75%的安全攻击发生在代码应用层面。

已知安全漏洞中，应用程序安全漏洞与 Web 应用程序安全漏洞占多数。美国国家标准技术研究院（NIST）的统计数据显示 92%的漏洞属于应用层而非网络层。国家计算机网络应急技术处理协调中心 2020 年 4 月的发布的数据显示，2019 年，国家信息安全漏洞共享平台（CNVD）收录的安全漏洞数量创下历史新高，数量同比增长 14.0%，达到 16193 个，其中应用程序漏洞占比 56.2%，Web 应用程序占比 23.3%，二者相加占比超过 76%，充分说明安全漏洞大多存在于软件应用服务本身。

传统研发运营安全模式中，安全介入相对滞后。传统研发运营安全，针对服务应用自身的安全漏洞检测修复，通常是在系统搭建或者功能模块构建完成之后以及服务应用上线运营之后，安全介入，进行安全扫描，威胁漏洞修复。如当前的大多数安全手段，防病毒、防火

墙、入侵检测等，都是关注软件交付运行之后的安全问题，属于被动防御性手段。这种模式便于软件应用服务的快速研发部署，但安全介入相对滞后，并无法覆盖研发阶段代码层面的安全，安全测试范围相对有限，安全漏洞修复成本也更大。

安全左移有助于帮助企业削减成本。代码是软件应用服务开发的最初形态，其缺陷或漏洞是导致安全问题的直接根源，尽早发现源码缺陷能够大大降低安全问题的修复成本。根据美国国家标准与技术研究所（NIST）统计，在发布后执行代码修复，其修复成本相当于在设计阶段执行修复的 30 倍。具体数据如图 2 所示。



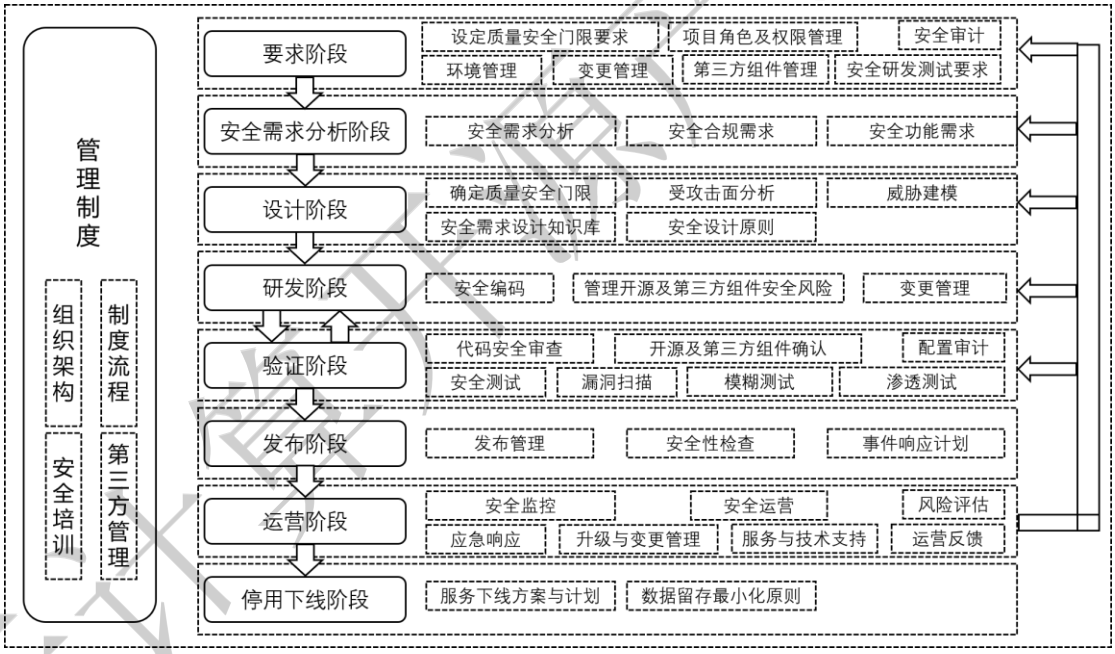
数据来源：美国国家标准与技术研究所（NIST）

图 2 研发运营各阶段代码漏洞修复成本

在此背景下，搭建新型的研发运营安全体系，进行安全左移，覆盖软件应用服务的全生命周期，是至关重要，也是势在必行的。建立新型的研发运营安全体系有助于构建可信理念，创造可信生态，是实现软件应用服务全生命周期安全的重要一步。

（二）覆盖软件应用服务全生命周期的研发运营安全体系

新型研发运营安全体系强调安全左移，覆盖软件应用服务全生命周期。本白皮书认为的研发运营安全指结合人员管理体系、制度流程，在软件应用服务设计早期便引入安全，进行安全左移，覆盖要求阶段、安全需求分析阶段、设计阶段、研发阶段、验证阶段、发布阶段、运营阶段、停用下线阶段的全生命周期，搭建安全体系，降低安全问题解决成本，全方面提升服务应用安全，提升人员安全能力。具体架构体系如下图 3 所示。



图片来源：中国信息通信研究院

图 3 研发运营安全体系

体系框架具体内容包括：1）管理制度，建立合适的人员组织架构与制度流程，保证研发运营流程安全的具体实施，针对人员进行安全培训，增强安全意识，进行相应考核管理；2）明确安全要求，前期明确安全要求，如设立质量安全门限要求，进行安全审计，对于第三

方组件进行安全管理等；3）安全需求分析与设计，在研发阶段之前，进行安全方面的需求分析与设计，从合规要求以及安全功能需求方面考虑，进行威胁建模，确定安全需求与设计；4）安全研发测试，搭配安全工具确保编码实际安全，同时对于开源及第三方组件进行风险管理，在测试过程中，针对安全、隐私问题进行全面、深度的测试；5）安全发布，服务上线发布前进行安全性审查，制定事先响应计划，确保发布安全；6）运营安全，上线运营阶段，进行安全监控与安全运营，通过渗透测试等手段进行风险评估，针对突发事件进行应急响应，并及时复盘，形成处理知识库，汇总研发运营阶段的安全问题，形成反馈机制，优化研发运营全流程；7）停用下线，制定服务下线方案与计划，明确隐私保护合规方案，确保数据留存符合最小化原则，满足国家相关规范要求。

二、 研发运营安全发展现状

（一）全球研发运营安全市场持续扩大

全球信息安全市场保持稳定增长，应用安全市场增速高于整体安全市场。本白皮书提出的研发运营安全强调安全左移，通过自动化安全测试工具，关注软件应用服务代码层面安全，与应用安全紧密关联。根据 Gartner 2020 年 6 月发布的统计数据显示，全球 2019 年各项安全类支出总计 1209.34 亿美元，预计 2020 年将达到 1238.18 亿美元，其中应用安全市场规模 2019 年为 30.95 亿美元，预计 2020 年将达到

32.87 亿美元，年增长率达到 6.2%，明显高于整体信息安全市场的 2.4%年增长率，具体数据如表 1 所示。

市场领域	2019	2020	增长率（%）
应用安全	3095	3287	6.2
云安全	439	585	33.3
数据安全	2662	2852	7.2
身份访问管理	9837	10409	5.8
基础设施保护	16520	17483	5.8
综合风险管理	4555	4731	3.8
网络安全设备	13387	11694	-12.6
其他信息安全软件	2206	2273	3.1
安全服务	61979	64270	3.7
客户安全软件	6254	6235	-0.3
总计	**120934**	**123818**	**2.4**

数据来源：Gartner，2020 年 6 月

表 1 2019-2020 全球各项安全类支出及预测（单位：百万美元）

我国应用安全市场增速高于全球，市场规模占全球比例达到近三分之一。2019 年，我国应用安全市场规模达到 8.48 亿美元，市场规模占全球应用安全市场规模比例达到近三分之一，预计 2020 年市场规模将达到 9.45 亿美元，年增长率达到 11.5%，高于全球 6.2%的增长率。具体数据如表 2 所示。

市场领域	2019	2020	增长率（%）
应用安全	848	945	11.5
云安全	1336	1448	8.3
数据安全	565	616	9
身份访问管理	1730	1852	7.1
基础设施保护	2139	2318	8.4
综合风险管理	97	106	9.9
网络安全设备	7518	7111	-5.4
其他信息安全软件	379	395	4.2
安全服务	13173	15078	14.5
客户安全软件	27784	29869	7.5
总计	848	945	11.5

数据来源：Gartner，2020 年 6 月

表 2 2019-2020 中国各项安全类支出及预测（单位：百万美元）

应用程序安全测试（AST）市场增速最为迅猛，市场规模占比超过应用安全总体市场规模的三分之一。根据 Gartner 2019 年 4 月发布的报告调查数据显示，应用安全测试市场预计将以 10%的复合年增长率增长，这仍是信息安全领域中快速增长的部分，到 2019 年底，AST 的市场规模估计将达到 11.5 亿美元，市场规模占比超过应用安全总体市场规模的三分之一。根据 Industry Research 2019 年 8 月发布的数据显示，按照应用程序安全测试类型区分，静态应用程序安全测试（SAST）将占主导地位，预计将以 24.06%的复合年增长率增长，交互式应用程序安全性测试（IAST）预计将以最快的 27.58%的复合年增长率增长。

（二）国家及区域性国际组织统筹规划研发运营安全问题

重点国家与区域性国际组织已发布政策规范，重视研发运营安全问题。软件应用服务是信息化的重要组成部分，源代码是软件应用服

务的最原始形态。越来越多的国家已经意识到软件应用服务的源代码安全的重要性，在强调安全运营、防御的基础之上，通过发布一系列政策以及指南，从国家层面规范此方面的工作。目前美国、英国、俄罗斯、印度、澳大利亚以及欧盟等国家和区域组织都已经推行涉及研发运营安全的战略、规范或指南，其中以美国、英国、印度、欧盟最为典型。

美国发布战略计划，关注研发运营安全。美国越来越依赖于网络空间，但安全并未跟上网络威胁的增长。**关于研发安全，**美国国家科技委员会（NSTC）网络和信息技术研发分委会在 2019 年 12 月发布《联邦网络安全研发战略计划》，主要内容涵盖四个相互关联的防御能力：威慑、保护、探测、响应。在威慑能力中明确提出，设计安全的软件是威慑手段之一；保护能力关于研发安全具体包含两个方面的内容，1）减少脆弱性，具体行为包括安全设计、安全开发、安全验证、可持续安全；2）执行安全原则，具体涵盖使用加密机制保护数据，提高访问控制效率，避免安全漏洞引入。此外，**美国国土安全部资助软件质量保证（SQA）项目，提升软件应用安全性，**软件质量保证（SQA）项目发展工具与技术，用于分析识别软件中的潜在安全漏洞，具体而言，该项目强调软件代码开发过程中，安全性分析以及脆弱性识别，从而在开发过程的早期发现并消除漏洞、缺陷。**关于运营安全，**《联邦网络安全研发战略计划》中提出的探测与响应能力和运营安全密切相关，具体内容包括实时监测系统安全，及时检测甚至预

测安全威胁，动态评估安全风险，对于安全威胁进行联动、自适应处理等内容。

英国推行源码审查，发布研发运营安全相关战略指南，提升整体软件应用服务安全性。英国基于自身 IT 产业情况，使用其他国家企业的网络信息技术产品和服务的情况较多，供应链更为复杂。**针对软件应用代码安全以及研发安全，**英国推行网络安全审查机制，采用相对市场化的评估机制，这其中就包括深层次的源代码审查测试，检测相关产品或服务是否存在安全缺陷或漏洞。同时，英国国家网络安全中心（NCSC）于 2018 年 11 月发布《安全开发和部署指南》，具体包含 8 项安全开发原则，1）安全开发关系每一个人；2）保持安全知识实时更新；3）研发干净可维护的代码；4）保护开发环境；5）保护代码库；6）保护构建和部署管道；7）持续进行安全性测试；8）对于安全威胁、漏洞影响提前计划，8 项原则均与研发安全密切相关。**针对运营安全，**《国家网络安全战略 2016-2021》中明确提出要提升防御能力，提高政府和公共部门抵御网络攻击的弹性，定期评估关键系统的安全漏洞，业界应与国家网络安全中心（NCSC）共享网络威胁最新情报，进行主动防御等具体举措。

印度推行国家战略，推动系统应用研发运营安全。印度作为软件开发大国，对代码安全方面的要求较高。**针对研发安全，**印度国家网络协调中心（NCCC）在 2013 年曾发布《国家网络安全战略》（NCSS 2013），推动网络安全研究与开发是其战略之一，包括解决与可信系统的开发、测试、部署和维护整个生命周期相关的所有问题。2020 年，

正在征求意见更新《国家网络安全战略》（NCSS 2020），安全测试左移，集成到开发周期之中，安全团队成为应用程序开发生命周期的一部分是主要趋势之一。**针对运营安全**，在战略中提出，创建安全威胁早期预警、漏洞管理和应对安全威胁的机制；建立国家级的系统、流程、结构和机制，对现有和潜在网络安全威胁进行必要情境推测。

欧盟颁布实施法案，注重国际合作，推进整体的研发运营安全。欧盟在 2019 年 6 月 27 日，正式施行《网络安全法案》，旨在有效应对随着数字化和连接性的增加而带来的与网络安全相关的各类风险，防范对计算机系统、通信网络、数字产品、服务和设备等带来的潜在威胁，进一步完善欧盟的网络安全保护框架。**针对研发设计安全**，明确提出，参与产品设计开发的组织、制造商、提供商应在设计和开发的早期阶段采取安全措施，推测安全攻击的发生，减少安全风险的影响，同时安全应贯穿产品全生命周期，以减少规避安全风险。**针对运营安全**，强调制定和更新联盟级别的网络和信息系统安全策略，对于安全事件联合处理，内部成员国共享安全信息、技术解决办法，提升事件处置的效率。

	美国	英国	印度	欧盟
国家政策、指南	• 《联邦网络安全研发战略计划》 • 软件质量保证（SQA）项目	• 网络安全审查机制 • 《安全开发和部署指南》	• 《国家网络安全战略》	• 《网络安全法案》
研发安全	• 《联邦网络安全研发战略计划》中涵盖威慑、保护、探测、响应四项能力，其中威慑、保护与研发安全密切相关，在威慑中明确提出，设计安全的软件是威慑手段之一；保护具体包含两个方面，1）减少脆弱性，具体行为包括安全设计、安全开发、安全验证、可持续安全；2）执行安全原则，具体涵盖使用加密机制保护数据，提高访问控制效率，避免安全漏洞引入； • 美国国土安全部资助软件质量保证（SQA）项目，提升软件应用安全性，软件质量保证（SQA）项目发展工具与技术，用于分析识别软件中的潜在安全漏	• 推行网络安全审查机制，采用相对市场化的评估机制，其中包括深层次的源代码审查测试，检测相关产品或服务是否存在安全缺陷或漏洞 • 英国国家网络安全中心（NCSC）发布《安全开发和部署指南》，包含 8 项安全开发原则，1）安全开发关系每一个人；2）保持安全知识实时更新；3）研发干净可维护的代码；4）保护开发环境；5）保护代码库；6）保护构建和部署管道；7）持续进行安全性测试；8）对于安全威胁、漏洞影响提	• 印度国家网络协调中心（NCCC）在 2013 年发布《国家网络安全战略》（NCSS 2013），推动网络安全研究与开发是其战略之一，包括解决与可信系统的开发、测试、部署和维护整个生命周期相关的所有问题。2020 年，正在征求意见更新《国家网络安全战略》（NCSS 2020），安全测试左移，集成到开发周期之中，安全团队成为应用程序开发生命周期的一部分是主要趋势之一。	• 2019 年 6 月 27 日施行的《网络安全法案》中明确提出，参与产品设计开发的组织、制造商、提供商应在设计和开发的早期阶段采取安全措施，推测安全攻击的发生，减少安全风险的影响，同时安全应贯穿产品全生命周期，以减少规避安全风险。

	洞，具体而言，该项目强调软件代码开发过程中，安全性分析以及脆弱性识别，从而在开发过程的早期发现并消除漏洞、缺陷。	前计划，与研发安全密切相关。		
运营安全	• 《联邦网络安全研发战略计划》提出的四项能力中，探测和响应与安全运营密切相关，具体内容包括实时监测系统安全，及时检测甚至预测安全威胁，动态评估安全风险，对于安全威胁进行联动、自适应处理。	• 《国家网络安全战略2016-2021》中明确提出要提升防御能力，提高政府和公共部门抵御网络攻击的弹性，定期评估关键系统的安全漏洞，业界与国家网络安全中心（NCSC）共享网络威胁最新情报，进行主动防御等具体举措。	• 《国家网络安全战略》中提出，创建安全威胁早期预警、漏洞管理和应对安全威胁的机制；建立国家级的系统、流程、结构和机制，对现有和潜在网络安全威胁进行必要情境推测等具体措施。	• 《网络安全法案》强调制定和更新联盟级别的网络和信息系统安全策略，对于安全事件联合处理，内部成员国共享安全信息、技术解决办法，提升事件处置的效率。

表 3 重点国家及区域性国际组织研发运营安全相关举措

（三）国际标准组织及第三方非盈利组织积极推进研发运营安全共识

ISO27304 关注建立安全软件程序流程和框架。ISO27034 是国际标准化组织通过的第一个**关注建立安全软件程序流程和框架**的标准，提供了面向企业落地应用安全生命周期的指导框架，其本质目的是指

导企业如何通过标准化的方式把安全融合进入软件生命周期。ISO27034 由七个部分组成，除了第四部分外已全部发布。

SAFECode 专注于应用安全。SAFECode 成立于 2007 年 10 月，在过去 10 余年中，其发布的指南已被用于为许多重要行业和政府提供信息，以解决软件安全问题。SAFECode 组织认为尽管存在差异，但业界公认的通用安全开发实践已被证明既实用又有效；在软件保证流程和实践中提供更高的透明度有助于客户和其他关键利益相关者有效地管理风险。2018 年 3 月 SAFECode 发布第三版《安全软件开发基本实践》，并在之后持续更新。《安全软件开发基本实践》说明保证软件安全的具体开发和实施细则，以确保软件按预期运营并且没有设计缺陷和实现缺陷，具体内容包括安全设计原则、威胁建模、安全编码实践、测试和验证、脆弱性及安全事件响应等。

OWASP（Open Web Application Security Project，即开放 Web 应用程序安全项目）关注软件安全，致力于改善软件的安全性，推动全球软件安全的革新与发展。OWASP 是一个非盈利组织，于 2004 年 4 月 21 日在美国成立。OWASP 提供有关计算机和互联网应用程序的公正、实际、有成本效益的信息，协助个人、企业和机构来发现和使用可信赖软件。其发布的 OWASP Top 10 代表了对 Web 应用程序最严重的 10 大安全风险，已经成为业界共识，是企业制定代码安全策略的重要参考文件，也是进行安全编码的有效一步。

	ISO27034	SAFECode	OWASP
关注对象	• 建立安全软件程序流程和框架	• 应用安全	• 软件安全，改善软件的安全性
具体内容	• ISO27034 由七个部分组成，是国际标准化组织通过的**第一个关注建立安全软件程序流程和框架的标准**，提供了面向企业落地应用安全生命周期的指导框架，其本质目的是指导企业如何通过标准化的方式把安全融合进入软件生命周期。	• SAFECode 发布的指南已被用于为许多重要行业和政府提供信息，以解决软件安全问题； • 其发布的**《安全软件开发基本实践》说明保证软件安全的具体开发和实施细则**，以确保软件按预期运营并且没有设计缺陷和实现缺陷，具体内容包括安全设计原则、威胁建模、安全编码实践、测试和验证、脆弱性及安全事件响应等。	• OWASP 提供有关计算机和互联网应用程序的公正、实际、有成本效益的信息，协助个人、企业和机构来发现和使用可信赖软件 • 其发布的**OWASP Top 10 代表了对 Web 应用程序最严重的 10 大安全风险**，已经成为业界共识，是企业制定代码安全策略的重要参考文件，也是进行安全编码的有效一步。

表 4 国际标准组织及第三方非盈利组织研发运营安全相关工作

（四）企业积极探索研发运营安全实践

微软持续改进安全开发生命周期(SDL,即 Security Development Lifecycle，以下简称 SDL）流程措施，推行研发运营安全实践。自 2004 年以来，SDL 作为微软一项强制性政策，在将安全性融入企业文化与软件开发实践中发挥了重大作用，在推出之后，对于其内容也在不断进行更新改进，目前具体举措包括 1）管理制度，提供安全培训，增强安全意识，确保人员了解安全基础知识；2）安全要求，定义安全隐私要求与安全门限要求，包括法律和行业要求，内部标准和编码

惯例，对先前事件的审查以及已知威胁等，安全门限要求指安全质量的最低可接受级别，明确定义安全漏洞的严重性阈值；3）安全隐私需求分析与设计，具体措施包括执行威胁建模，建立设计要求，明确加密标准；4）管理使用第三方组件的安全风险，拥有准确的第三方组件清单，并了解其安全漏洞可能对集成它们的系统的安全性产生影响；5）安全研发与验证，具体举措包括使用经过安全性检查，认可的工具，执行静态应用程序与动态应用程序安全性测试，进行渗透测试；6）发布部署阶段，建立标准的事件响应流程，7）针对运营安全，通过人员权限认证，数据加密存储、传输，安全监控，定期更新安全策略，抵御常见网络攻击，执行渗透测试等手段保证上线运营阶段的安全。

借鉴行业领先实践、技术，思科推行企业安全开发生命周期，减少产品安全风险。如下图 4 所示，具体措施涵盖，1）明确安全要求，包括思科内部安全基线要求与基于行业、场景的市场安全要求；2）第三方安全，利用工具了解潜在的第三方软件安全威胁，不断更新已知第三方软件威胁和漏洞列表，对于产品团队进行告警；3）安全需求分析与设计，内部安全培训计划鼓励所有员工提高安全意识，同时鼓励开发和测试团队深入学习安全知识，通过持续不断地发展威胁意识，并利用行业标准原则和高度安全的经过审查的解决方案，努力开发出设计上更加安全的产品；通过威胁建模了解和确定系统的安全风险并确定优先级；4）安全编码与验证，建立内部的安全编码标准，维护经过审核的通用安全模块，同时通过静态应用程序安全测试与漏

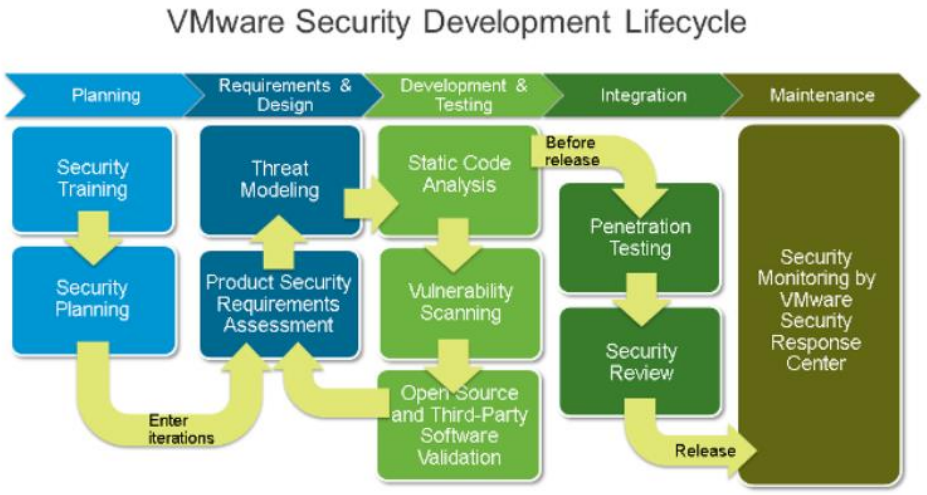
洞扫描，渗透测试等手段保证安全性；5）发布部署，建立安全发布标准，确保发布部署安全；6）上线运营，通过安全运营操作流程与持续安全监控、更新保证产品上线之后的安全。

图片来源：Cisco

图 4 Cisco SDL 体系框架图

VMware 建立安全开发生命周期项目，识别和减少 VMware 软件产品研发阶段的安全风险。VMware SDL 的发展受到行业最佳实践和组织的深远影响，会定期评估 SDL 在识别风险方面的有效性，并随着 SDL 活动的发展和成熟不断对于流程进行优化改进，增添新型技术。目前 VMware SDL 中的安全活动主要囊括，1）安全培训，对于人员进行基于角色和技术的安全以及隐私培训；2）安全规划，对于随后的安全审查活动制定基线要求；3）安全要求，涵盖认证、授权、加密、证书、网络安全性等内容；4）安全设计，通过威胁建模明确安全风险，改进安全设计；5）安全研发验证，通过静态代码分析与漏洞扫描、渗透测试等手段保证研发验证阶段的安全性；6）开源及第三方

组件安全管理，明确产品中开源及第三方组件中的安全漏洞，在发布前期进行修复；7）安全审查，对于前期所有安全工作进行二次审查；8）上线运营，由安全响应中心进行持续的安全监控，对于安全风险进行及时响应。具体执行流程如下图 5 所示。



图片来源：VMware

图 5 VMware SDL 体系框架图

		微软	思科	VMware
软件应用服务研发运营全生命周期安全管理	安全培训	• 提供安全培训，增强安全意识，确保人员了解安全基础知识	• 内部安全培训计划鼓励所有员工提高安全意识，同时鼓励开发和测试团队深入学习安全知识	• 对于人员进行基于角色和技术的安全以及隐私培训
	安全要求	• 定义安全隐私要求与安全门限要求，包括法律和行业要求，内部标准和编码惯例，对先前事件的审查以及已知威胁等	• 包括思科内部安全基线要求与基于行业、场景的市场安全要求	• 对于之后的安全审查活动制定基线要求 • 安全要求涵盖认证、授权、加密、证书、网络安全性等内容
	安全隐私需求分析与设计	• 执行威胁建模，建立设计要求，明确加密标准等	• 利用行业标准原则和高度安全的经过审查的解决方案，努力开发出设计上更加安全的产品 • 通过威胁建模了解和确定系统的安全风险并确定优先级	• 通过威胁建模明确安全风险，改进安全设计
	第三方组件安全管理	• 拥有准确的第三方组件清单，并了解其安全漏洞可能对集成它们的系统的安全性产生影响	• 利用工具了解潜在的第三方软件安全威胁，不断更新已知第三方软件威胁和漏洞列表，对于产品团队进行告警	• 明确产品中开源及第三方组件中的安全漏洞，在发布前期进行修复
	安全编码与验证	• 使用经过安全性检查，认可的工具，执行静态应用程序与动态应用程序安全性测试，进行渗透测试	• 建立内部的安全编码标准，维护经过审核的通用安全模块，同时通过静态应用程序安全测试与漏洞扫描，渗透测试等手段保证安全性	• 通过静态代码分析与漏洞扫描、渗透测试等手段保证研发验证阶段的安全性
	安全发布部署	• 建立标准的事件响应流程	• 建立安全发布标准，确保发布部署安全	• 对于前期所有安全工作进行二次审查

	上线运营安全	• 通过人员权限认证，数据加密存储、传输，安全监控，定期更新安全策略，抵御常见网络攻击，执行渗透测试等手段保证上线运营阶段的安全	• 通过安全运营操作流程与持续安全监控、更新保证产品上线之后的安全	• 由安全响应中心进行持续的安全监控，对于安全风险进行及时响应

表 5 企业研发运营安全具体实践

（五）开发模式逐步向敏捷化发展，研发运营安全体系随之向敏捷化演进

研发运营安全相关体系的发展与开发模式的变化是密不可分的，随着开发模型由传统的瀑布式开发演变成敏捷开发再转变为 DevOps，研发运营安全相关体系也随着变化，但其核心理念始终是安全前置，贯穿全生命周期。目前研发运营安全体系中，以微软提出的安全开发生命周期（SDL）和 Gartner 提出的 DevSecOps 体系为典型代表。

安全开发生命周期（SDL）的核心理念就是将安全考虑集成在软件开发的每一个阶段:需求分析、设计、编码、测试和维护。从需求、设计到发布产品的每一个阶段每都增加了相应的安全活动，以减少软件中漏洞的数量并将安全缺陷降低到最小程度。安全开发生命周期(SDL)是侧重于软件开发的安全保证过程，旨在开发出安全的软件应用。SDL 在传统软件开发生命周期(SDLC)的各个阶段增加了一些必要的安全活动，软件开发的不同阶段所执行的安全活动也不同，每个活

动就算单独执行也都能对软件安全起到一定作用。具体内容如下图 6 所示。

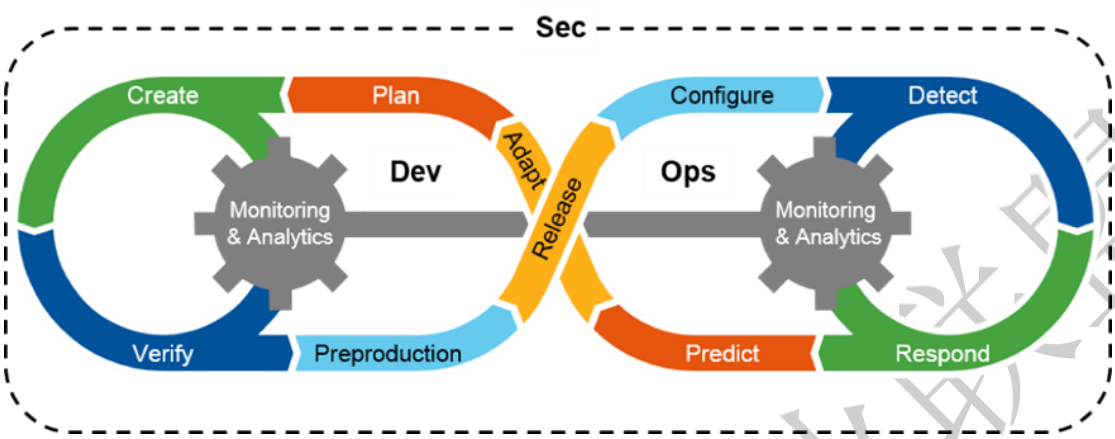
图片来源：Microsoft

图 6 微软 SDL 流程体系

随着对软件开发质量和效率要求的不断提高，以 DevOps 为代表的敏捷开发方法得到推崇。在此基础上，Gartner 公司于 2012 年推出了 DevSecOps 的概念，DevSecOps 即 Development Security Operations 的缩写，是一套基于 DevOps 体系的全新安全实践战略框架，旨在将安全融入敏捷过程中，即通过设计一系列可集成的控制措施，增大监测、跟踪和分析的力度，优化安全实践，集成到开发和运营的各项工作中，并将安全能力赋给各个团队，同时保持“敏捷”和“协作”的初衷。

DevOps 的目的决定了其对“自动化”和“持续性”的要求更加突出，因此在将安全控制集成其中时，也应该尽量遵循“自动化”和“透明”的原则。为了将安全无缝集成到 DevOps 中，Gartner 和一些专家从实践出发提出了一系列建议，主要包括：风险和威胁建模、自定义代码扫描、开源软件扫描和追踪、考虑供应链安全问题、整合预防性安全控制到共享源代码库和共享服务中、版本控制和安全测试的自动

化部署、系统配置漏洞扫描、工作负载和服务的持续监控等。下图 7 为 DevSecOps 具体体系框架。



图片来源：Gartner

图 7 DevSecOps 体系框架图

	SDL	DevSecOps
适用对象	软件产品安全开发全生命周期	DevOps 体系，周期较短、迭代较快的业务
安全责任	特定安全团队	研发运营所有参与人员
体系特点	安全集成在软件开发的每一个阶段，整体提升安全性	DevOps 体系中融入安全，安全工具自动化以及平台化
体系重点	整体安全管理制度搭配安全人员能力达到软件产品研发安全	DevOps 体系中嵌入自动化安全工具，实现 DevOps 体系的安全

表 6 SDL 与 DevSecOps 区别对照

三、 研发运营安全关键要素

本白皮书认为的研发运营安全关键要素包含两方面内容，1）**覆盖软件应用服务全生命周期的研发运营安全体系，**提供理论框架，指导研发运营安全的实践推进；2）**研发运营安全技术工具的持续发展**

应用，为体系的实践提供技术支撑，加速企业组织研发运营安全的落地。

（一）覆盖软件应用服务全生命周期的研发运营安全体系

本白皮书提出的研发运营安全体系强调安全左移，结合人员管理体系、制度流程，从需求分析设计、编码阶段便引入安全，覆盖软件应用服务全生命周期，整体提升安全性。提出的研发运营安全体系具有四大特点，1）覆盖范围更广，延伸至下线停用阶段，覆盖软件应用服务全生命周期；2）更具普适性，抽取关键要素，不依托于任何开发模式与体系；3）不止强调安全工具，同样注重安全管理，强化人员安全能力；4）进行运营安全数据反馈，形成安全闭环，不断优化流程实践。

1. 管理制度

管理制度流程是推行研发运营安全的基础。在研发运营安全体系规划和建设的过程中，首先是建立组织责任体系，制定完善的研发运营安全管理体系和制度管理规范，明确管理制度和操作流程规范，建立统一的安全基线。并将组织建设和人员制度管理纳入到全生命管理周期中，对应的组织负责不同的安全职责与工作，进行安全培训，建设组织级的安全文化以及对研发人员、测试人员、技术运营人员等进行安全管理，包括第三方机构的人员，实现人人都为安全负责。

制度和操作规范包括1)账号和密码管理,2)故障流程管理办法，3）应急事件分级处理措施，4）人员行为安全规范，5）变更管理制度，6）团队间安全协作流程和规范等。通过统一的流程管理平台，保证各个流程环节能够被及时响应,各项任务能够被顺利传递、衔接。

安全培训针对所有研发、测试、运营人员以及第三方合作人员，目前是为了提升安全意识，增强研发运营安全能力。培训内容主要包括1）安全管理制度，2）安全意识培训，3）安全开发流程，4）安全编码规范,5)安全设计,6)安全测试等，并对于培训结果进行考核，制定特殊岗位的上岗前考核机制，未通过相关考核的，不得从事向相关岗位的工作。

2. 安全要求

安全要求明确研发运营安全的基线。安全要求通常包含安全管理和技术安全要求，二者需要有机结合，不可分割。具体内容包括 1）设立质量安全门限要求，具有项目级、团队级、组织级的质量安全门限要求,根据业务场景、产品类型、语言类型划分质量安全门限要求，智能化收集质量安全门限要求，根据业务场景等进行智能推荐；2）项目角色以及权限管理，依据最小权限原则，建立资源、行为操作权限管控，采用多因素认证机制保证访问安全，配置强密码策略，及时为不需要权限的用户或用户组移除权限；3）安全审计，对于包括研发、测试、运营的所有相关人员的所有操作行为进行审计，对于审计记录进行保护，有效期内避免非授权的访问、篡改、覆盖及删除，对于审计记录形成报表，方便查询、统计与分析，针对审计日志进行自

动化与人工审计，对于安全事件进行详细记录，对于高危操作进行重点审计，进行告警通知，针对行业特点，业务特点进行定制化的安全审计策略，对于审计记录进行统计分析、关联分析等；4）环境管理，研发、测试、生产环境隔离，生产环境具有安全基线要求，保障环境的安全，针对研发、测试环境有明确的权限管控机制，针对各类环境的操作进行详细记录，具有可追溯性，定期执行生产环境的安全基线扫描，及时发现和处理安全风险，研发、生产环境具有良好的抗攻击与灾备容错能力，根据特定行业以及业务场景，对于测试环境接入安全扫描，针对不同的业务场景以及架构，对于发布环境进行分类管理，安全加固，生产环境具安全风险自动发现、分析和修复以及秒级容灾容错切换能力；5）变更管理，有明确的进行变更条件与变更执行机制，有明确的变更授权机制，对于变更请求进行统一分析、整理，确定变更方案；6）开源及第三方组件管理，具有组织级的第三方组件库，明确优选、可用、禁用的第三方组件列表，统一组件来源，具有明确的第三方组件入库审批机制，第三方组件的引入应遵循最小化引入原则，减少安全风险，开源及第三方组件与自研代码独立存放、目录隔离，开源及第三方组件来源可追溯，开源组件追溯源社区，第三方组件信息追溯到供应商，对开源软件的生命周期进行管理，记录开源软件的生命周期信息，通过自动化工具及时向使用产品进行通知预警；7）安全研发测试要求，具有组织级、团队级、项目级的安全编码规范、安全测试规范。

3. 安全隐私需求分析与设计

安全前置到需求分析与设计阶段。安全隐私需求分析与设计是服务应用研发运营整个生命周期的源头。具体内容包括：1）安全隐私需求分析，应包括安全合规需求以及安全功能需求，针对安全合规需求，应分析涉及的法律法规、行业监管等要求，制定合规和安全需求基线，针对安全功能需求应根据业务场景、技术，具备相应的测试用例，安全隐私需求来自法律法规、行业监管要求、公司安全策略、业界最佳实践以及客户安全需求，具有明确的安全需求管理流程，能够对安全需求的分析、评审、决策等环节进行有效管理，需求分解分配可追溯；2）安全设计原则，3）确定质量安全门限要求，4）受攻击面分析，分析应从系统各个模块的重要程度、系统各个模块接口分析、攻击者视角分析攻击手段、方式、攻击路径、权限设置是否合理、攻击难度等维度进行分析；5）威胁建模，具体行为包括确定安全目标、分解应用程序、确定威胁列表；6）安全隐私需求设计知识库，具有组织级安全需求知识分享平台，形成知识的复用，根据安全需求，得出安全设计解决方案。

4. 研发与验证

研发验证是安全前置实践的关键所在，研发阶段安全是整体安全左移实现关键，关注代码程序安全，验证阶段进行安全二次确认，避免风险引入。为了确保上线服务应用没有安全风险，需要在研发及测试过程中要进行全面的代码安全识别，具体内容包括：1）安全编码，维护获得安全认可的工具、框架列表，使用获得认可的工具、框架，具有统一的版本控制系统，将全部源代码纳入版本控制系统管理，版

本控制系统具有明确的权限管控机制，代码仓库具有实时代码安全扫描机制，发现安全问题并提示修复，根据安全编码规范制定自定义安全策略，进行自动化安全扫描，采用集成于 IDE 或其他形式提供的自动化测试工具定时进行代码安全检测，针对版本控制系统有监控机制，包括人员、时间、行为操作等，方便审计回溯，制定代码合入门禁机制，确保代码合入质量，代码仓库支持线上代码动态扫描，发现安全问题并提示修复；2）管理开源及第三方组件安全风险，对于第三方组件根据风险级别，有明确的优选、可用、禁用机制，代码提交前采用扫描工具进行第三方组件安全检查，管理项目中的第三方组件许可证以及安全漏洞等风险，针对第三方组件安全风险，推荐安全解决方案；3）变更管理，对于变更操作进行统一管理，明确记录变更信息，包括但不限于变更人员、变更时间、变更内容，针对重点变更内容进行评审，变更操作具有明确的审批授权机制，重大变更操作具有分级评审机制，具有统一的变更管理系统，变更操作覆盖需求设计到发布部署全流程；4）代码安全审查，制定明确的源代码安全检视方法，开展源代码安全审计活动，采用工具与人工核验相结合的方式进行代码安全审计，对于威胁代码及时通知研发人员进行修复，对高风险源代码有分级审核机制，对于审计发现的威胁代码自动通知研发人员，进行修复，根据行业特点、业务场景定制化开发代码安全审查工具，制定安全审查策略；5）开源及第三方组件确认，采用工具与人工核验的方式确认第三方组件的安全性、一致性，根据许可证信息、安全漏洞等综合考虑法律、安全风险；6）配置审计，具有明确的配置审

计机制，配置审计包括但不限于配置项是否完备、配置项与前期安全需求的一致性、配置项版本的描述精确，与相关版本一致制，配置项的每次变更有记录，可以追溯到具体修改时间和修改人，产品依赖的自研模块、平台组件、开源源码、开源二进制、第三方软件被准确的定义和记录，对于明确统一的合规需求以及安全需求，进行自动化配置审计；7）安全隐私测试，具有明确的安全隐私测试要求，作为发布部署的前置条件，测试数据不包含未经清洗的敏感数据，基于安全隐私需求，有相应的安全隐私测试用例，并进行验证测试，单个测试用例的执行不受其他测试用例结果的影响，测试数据、用例应统一管理，有明确的权限管控机制，测试用例、测试数据应定期更新，满足不同阶段、环境的测试要求，具备自动化安全测试能力，对于测试结果有集中汇总与展示的能力，持续优化安全测试策略，持续降低误报率与漏报率，测试过程有记录可查询，测试设计、执行端到端可追溯，基于不同业务场景以及系统架构，进行安全测试智能化推荐与测试策略智能优化；8）漏洞扫描，采用主流的安全工具进行漏洞扫描，漏洞扫描结果有统一管理与展示平台，漏洞扫描的结果及时反馈研发人员，进行漏洞修复，具有自身以及第三方漏洞库，对于漏洞库定期更新，基于漏洞信息进行关联与聚合分析；9）模糊测试，采用主流的模糊测试工具，自动化进行模糊测试，模糊测试的结果及时反馈研发人员，进行修复，持续改进模糊测试策略；10）渗透测试，引入人工渗透测试机制，针对系统架构、应用程序、网络层面漏洞进行渗透测

试，根据行业特点与业务场景实施渗透测试，范围应覆盖重要安全风险点与重要业务系统，有明确的渗透测试计划与管理机制。

5. 发布部署

安全发布部署是服务应用上线前的最后一道安全保障，发布阶段确保服务安全上线运营，具体内容包括：1）发布管理，有相应的发布安全流程与规范，发布操作具有明确的权限管控机制，发布应具有明确的安全检查节点，根据安全节点检查结果，有相关告警机制，针对发布流程具有安全回退、备份机制，制定发布策略，通过低风险的发布策略进行发布，如灰度发布或者蓝绿发布等方式，发布流程实现自动化，一键发布，根据安全节点检查结果，发现高危安全问题，自动阻断发布流程，对于发布流程具有监控机制，出现问题自动化回滚，建立稽核机制，发布前需要通过稽核部门的独立检查；2）安全性检查，进行病毒扫描以及数字签名验证等完整性校验，校验结果作为发布的前置条件；3）事件响应计划，具有预先的事件响应计划，包括但不限于安全事件应急响应流程，安全负责人与联系方式。

6. 上线运营

运营阶段安全保障服务系统的稳定运行。为确保服务应用上线运营安全，具体措施内容包括：1）安全监控，具有运营阶段安全监控机制，覆盖全部业务场景，抵御常见威胁攻击的能力，如 DDoS 攻击，暴力破解，病毒攻击，注入攻击，网页篡改，具有统一的安全监控平台，对于威胁攻击处理能够统一监控并可视化展示，对于监控安全事

件进行分级展示，具有智能化安全监控平台，对于监控事件统一关联分析，智能识别潜在的安全风险，实现智能化用户行为分析以及资产数据的安全画像；2）安全运营，定期进行常规安全检查与改进，运营人员有明确的权限管控机制与管理规范，监控运营数据加密存储，存储与备份机制符合安全要求，保证全生命周期安全，对于安全事件有多种方式的告警机制，通过统一平台对于安全事件处置全流程进行跟踪，具备从外部接收相关漏洞通告和安全情报的能力，对于自动化运维工具进行安全加固并具备自动化监控机制，及时发现工具的操作安全风险，对于运营过程中的安全日志等数据进行自动化分析，发现安全风险并告警，可建设统一的安全运营中心，分布于不同位置的云平台接入统一运营中心，将管理数据统一进行处理，对于监控数据进行统计、展示，具备持续的安全漏洞、安全信息外部反馈机制，对于运营过程中的安全日志等数据进行智能化关联分析，发现潜在安全风险并告警，根据漏洞信息、业务场景等智能化推荐安全解决方案，进行智能化处置；3）风险评估，制定和实施安全风险评估计划，定期进行安全测试与评估，安全风险评估、测试范围应覆盖重要业务系统、应用，建立渗透测试流程，根据渗透测试流程，针对系统架构、应用程序、网络层面漏洞进行安全测试，制定漏洞奖励计划，鼓励第三方渗透测试，安全风险评估、测试范围应覆盖全业务系统，建立智能化的风险评估体系，对于生产环境中的安全风险进行分析、告警；4）应急响应，具有明确的应急事件响应流程，基于应急事件进行分级、分类处理，具备专门的应急响应安全团队，有统一的技术平台，对于

应急事件进行全流程跟踪与可视化展示，对于应急事件及时复盘，形成相关处理知识库，对于应急事件处理具有具体的量化指标，包括但不限于威胁处理时间、响应时间，定期开展应急事件演练，对于应急响应事件可以实现一定程度上的自动化处理，对于应急事件具有全面的自动化以及一定程度智能化处理能力；5）升级与变更管理，有明确的升级与变更操作制度流程，升级变更操作有明确的权限管控机制，升级变更操作有明确的审批授权机制，对于升级变更操作有明确的操作信息记录，包括但不限于变更升级内容、变更升级时间，变更升级操作对于用户无感知，对于用户有影响的，需要提前告知沟通，有相应的回滚机制，变更升级操作与版本系统同步，确保版本信息一致，对于重大变更升级有分级评审机制，实现自动化变更升级与回滚，变更升级操作有相应监控机制，出现问题自动化回滚；6）服务与技术支持，有明确的服务与技术支持方式，通过电话等方式对于用户反馈问题进行反馈、回访，对于监管部门、运营商提出的安全问题及时响应，对于用户反馈问题有分级处理机制，及时对于处理结果进行反馈，说明处理结果、影响程度等，对于反馈问题分类处理、记录、归档，方便知识的反馈、复用，针对安全类问题具有专属反馈通道，确保安全问题的及时响应；7）运营反馈，定期收集运营过程中的安全问题，进行反馈，对于反馈安全问题分类、分级处理，完善前期安全需求设计、安全研发等流程，具有明确的反馈改善管理流程与度量机制，有统一的运营安全问题反馈平台，统一收集反馈的安全问题，分类、分级处理，反馈全流程跟踪，对于收集的安全问题自动化实现汇总分析，

优化从需求设计到研发运营整个流程，对于反馈安全问题实现智能化关联分析，发现潜在安全问题，优化研发运营全流程。

7. 停用下线

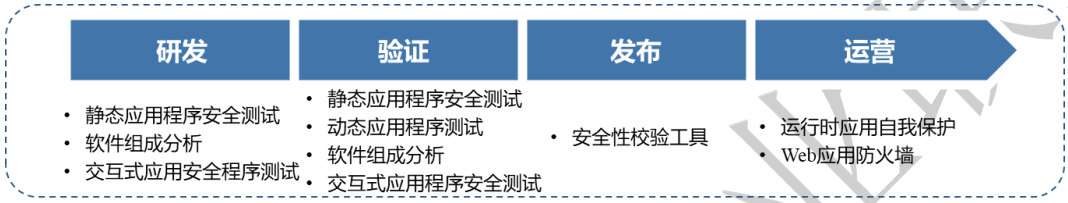
软件应用服务停用下线阶段安全实现研发运营安全体系闭环。服务停用下线是研发运营安全体系的最后一环，指系统在终止服务之后，应制定制定服务下线方案与计划，保护用户的隐私安全与数据安全，具体要求为明确隐私保护合规方案，确保数据留存符合最小化原则，满足国家相关规范要求。

（二）研发运营安全解决方案同步发展

研发运营安全的实践落地离不开自动化安全技术工具的持续发展。传统研发运营模式中，研发与安全割裂，主要是因为安全影响研发效率，通过自动化安全工具、设备，将安全融入软件应用服务的全生命周期，适应当前的敏捷开发等多种模式是实现研发运营安全的必要途径。同时，研发运营安全解决方案关注痛点安全问题，如安全要求、合规要求以及目前热点的个人数据和隐私保护等问题，使用安全解决方案可以更好的避免此类安全问题的发生，提升软件应用服务的安全性。

本白皮书中涉及的研发运营安全相关技术工具主要包括：研发验证阶段的静态应用程序安全测试（Static Application Security Testing ，以下简称 SAST），动态应用程序安全测试（Dynamic Application Security Testing，以下简称 DAST），交互式应用程序

安全测试（Interactive Application Security Testing，以下简称IAST），软件组成分析（Software Composition Analysis，以下简称 SCA）以及安全运营阶段的实时应用自我保护（Runtime Application Self-protection，以下简称 RASP）和 Web 应用防火墙（Web Application Firewall，以下简称 WAF）。下图 8 为具体对应阶段说明。



图片来源：中国信息通信研究院

图 8 研发运营安全解决方案阶段对应图

1. 静态应用程序安全测试

静态应用程序安全测试（SAST）是指不运行被测程序本身，仅通过分析或者检查源程序的语法、结构、过程、接口等来检查程序的正确性。源代码静态分析技术的发展与编译技术和计算机硬件设备的进步息息相关，源代码安全分析技术多是在编译技术或程序验证技术的基础上提出的，利用此类技术能够自动地发现代码中的安全缺陷和违背安全规则的情况。目前主流的分析技术包括：1）词法分析技术，只对代码的文本或 Token 流与已知归纳好的缺陷模式进行相似匹配，不深入分析代码的语义和代码上下文。词法分析检测效率较高，但是只能找到简单的缺陷，并且误报率较高。2）抽象解释技术，用于证明某段代码没有错误，但不保证报告错误的真实性。该技术的基本原理是将程序变量的值映射到更加简单的抽象域上并模拟程序的执行

情况。因此，该技术的精度和性能取决于抽象域对真实程序值域的近似情况。3）程序模拟技术，模拟程序执行得到所有执行状态，分析结果较为精确，主要用于查找逻辑复杂和触发条件苛刻的缺陷，但性能提高难度大。主要包括模型检查和符号执行两种技术，模型检查将软件构造为状态机或者有向图等抽象模型，并使用模态/时序逻辑公式等形式化的表达式来描述安全属性，对模型遍历验证这些属性是否满足；符号执行使用符号值表示程序变量值，并模拟程序的执行来查找满足漏洞检测规则的情况。4）定理证明技术，将程序错误的前提和程序本身描述成一组逻辑表达式，然后基于可满足性理论并利用约束求解器求得可能导致程序错误的执行路径。该方法较为灵活性，能够使用逻辑公式方便地描述软件缺陷，并可根据分析性能和精度的不同要求调整约束条件，对于大型工业级软件的分析较为有效。5）数据流分析技术，数据流分析技术基于控制流图，按照某种方式扫面控制流图的每一条指令，试图理解指令行为，以此判断程序中存在的威胁漏洞。数据流分析的通用方法是在控制流图上定义一组方程并迭代求解，一般分为正向传播和逆向传播，正向传播就是沿着控制流路径，状态向前传递，前驱块的值传到后继块；逆向传播就是逆着控制流路径，后继块的值反向传给前驱块。

静态应用程序安全测试（SAST）使用功能主要包括：1）代码提交、集成，在用户 IDE 环境中集成代码静态检测插件进行代码检查，对发现的问题能够直接显示在 IDE 上。将代码提交到远程仓库时，触发代码检查，系统会先去拉取代码，再执行代码扫描，扫描时支持全

量与增量代码扫描方式。构建时，支持每日定时设置，每天执行代码检查，同时也支持根据需要手动触发检查。2）风险展示与处理，扫描结束后，会在代码扫描系统及代码仓库上展示整体风险趋势变化，包括新增、修复、遗留告警、项目代码质量状态展示等，同时支持各个告警详情查看，包括风险等级、错误代码片段、问题描述、修复指导等。支持用户对告警进行标记，包括确认、误报、设计如此等多种场景。3）可扩展性，支持用户自定义规则扫描，也支持单个告警/批量告警/路径屏蔽等功能，便于提高与业务场景的契合度。

目前静态应用程序安全测试（SAST）主要厂商包括国外的Synopsys、Checkmarx、Veracode 等，以及国内的奇安信、默安科技等。

2. 软件组成分析

软件组成分析（SCA）主要针对开源组件，通过扫描识别开源组件，获取组件安全漏洞信息、许可证等信息，避免安全与法律法规风险。开源软件相比于闭源软件，带来了如获取便捷、降低成本等诸多好处，但相比于闭源软件，由于开源软件的所有权和使用权分离，如果使用不当，会导致最终的使用用户被迫承受风险。因此在使用中用户仍然需要注意遵循相关规则，例如遵循开源许可证的相关要求和监管条例、甚至需要公开自有的商业代码等，对引入和使用过程中潜在的安全风险进行有效监管。

目前来看，开源软件主要涉及的安全风险为以下三点：不清楚具体引用或使用了哪些开源组件、对开源许可证引入的知识产权和合规风险、开源软件自身的安全风险。

现有的开源扫描技术分为五种，1）通过进行源代码片段式比对来识别组件并识别许可证类型；2）对文件级别提取哈希值，进行文件级哈希值比对，若全部文件哈希值全部匹配成功则开源组件被识别；3）通过扫描包配置文件读取信息，进行组件识别从而识别组件并识别许可证类型；4）对开源项目的文件目录和结构进行解析，分析开源组件路径和开源组件依赖；5）通过编译开源项目并对编译后的开源项目进行依赖分析，这种方式可以识别用在开源项目中的开源组件信息。

上述 5 种识别技术的识别速度是依次增快的，并且组件物料清单的完整性也是依次增高的。源代码片段识别出的开源组件的数量较多，但因为源代码片段比对受行数和关键词位置影响，识别出的开源组件的误报率通常较高，且识别出的开源组件需要手动确认，对操作人员的技术能力要求较高；其他 3 类识别出的开源组件数量通常少于源代码片段识别，但因为哈希值的不变性，其识别出的开源组件的误报率较低，同时相比于开源代码片段识别，由于源代码被改写生成哈希值也会随之改变，因此漏报率通常比源代码片段识别高。

主流的安全漏洞检测原理为两种，第一种方法是依据获取到的开源组件名称和版本号信息，在公开的 CVE 或 CNVD 库里去查寻该版本曾经出现过的漏洞；第二种方法是通过程序分析技术，获取到开源组

件名称、版本信息和引用的函数，依据企业的商业漏洞库去匹配所引用的函数是否会造成漏洞。方法二的准确性远远高于方法一，但是实现难度也非常大。

针对开源组件自身安全风险，与传统的软件漏洞修复流程不同的是并不对开源软件做漏洞修补工作，开源软件漏洞治理通常会依靠扫描技术发现存有安全漏洞的开源软件版本号，与当前最新版本号做匹配，进行替换。因此开源软件的版本号管理、漏洞更新及跟踪工作也十分重要。

目前软件组成分析（SCA）主要厂商包括国外的 Synopsys、Micro Focus、WhiteHat Security 等，以及国内的奇安信、棱镜七彩等。

3. 交互式应用程序安全测试

交互式应用程序安全测试（IAST）是 2012 年 Gartner 公司提出的一种新的应用程序安全测试方案，通过代理和在服务端部署的 Agent 程序，收集、监控 Web 应用程序运行时请求数据、函数执行，并与扫描器端进行实时交互，高效、准确的识别安全漏洞，同时可准确确定漏洞所在的代码文件、行数、函数及参数。

交互式应用程序安全测试（IAST）主要在三方面做工作：流量采集、Agent 监控、交互扫描。1）流量采集，指采集应用程序测试过程中的 HTTP/HTTPS 请求流量，采集可以通过代理层或者服务端 Agent。采集到的流量是测试人员提交的带有授权信息有效数据，能够最大程度避免传统扫描中因为测试目标权限问题、多步骤问题导致扫描无效；同时，流量采集可以省去爬虫功能，避免测试目标爬虫无法爬取到导

致的扫描漏水问题。2）Agent 监控，指部署在 Web 服务端的 Agent 程序，一般是 Web 服务编程语言的扩展程序，Agent 通过扩展程序监控 Web 应用程序性运行时的函数执行，包括 SQL 查询函数、命令执行函数、代码执行函数、反序列化函数、文件操作函数、网络操作函数，以及 XML 解析函数等有可能触发漏洞利用的敏感函数。3)交互扫描，指 Web 应用漏洞扫描器通过 Agent 监控辅助，只需要重放少量采集到的请求流量，且重放时附带扫描器标记，即可完成对 Web 应用程序漏洞的检测。例如在检测 SQL 注入漏洞时，单个参数检测，知名开源 SQL 注入检测程序 SQLMAP 需要发送上千个 HTTP 请求数据包；交互扫描只需要重放一个请求，附带上扫描器标记，Agent 监控 SQL 查询函数中的扫描器标记，即可判断是否存在漏洞，大大减少了扫描发包量。

目前交互式应用程序安全测试（IAST）主要厂商包括国外的 Synopsys、Veracode 等，以及国内的悬镜安全、默安科技等。

4. 动态应用程序安全测试

动态应用程序安全测试（DAST）技术在测试或运行阶段分析应用程序的动态运行状态。它模拟黑客行为对应用程序进行动态攻击，分析应用程序的反应，从而确定该应用是否易受攻击。

以 Web 网站测试为例对于动态应用程序安全测试进行介绍，主要包括三个方面的内容：1）信息收集，测试人员在测试开始前，需要收集待测试网站的全部 URL，包括静态资源和动态接口等，每一条 URL 需要包含路径和完整的参数信息。测试人员收集 URL 的方式包括但不限于：（1）从网站源代码中获取 URL 的路径和参数信息，（2）编写

网络爬虫对目标网站进行爬取，获得网站每一个页面中包含的全部URL信息。网络爬虫不仅要具备静态页面的爬取和分析能力，同时也要具备对Web2.0时代新型的动态展示页面的爬取和分析能力。此外，网络爬虫在最终输出结果前，应当具备以某种规则对URL列表进行去重的能力，（3）在目标网站的服务器上安装Web流量采集工具，该工具会记录该网站所有的Web请求。测试人员模拟正常访问把目标网站的所有功能都是访问一遍，随后从流量采集工具中导出全部请求信息，从中提取出网站的URL列表。除了收集URL，测试人员还要解决目标网站访问权限的问题。如果网站的部分功能需要账号登录后才能访问，则测试人员需准备一套或多套能满足测试要求的账号密码信息，并将信息传递到安全测试工具中，且保证测试过程产生的数据包都有携带登录态信息。2）测试过程，测试开始前，测试人员应当将测试所需的URL列表导入到测试工具中。导入的方式包括但不限于：（1）手工从测试工具的管理页面逐条添加，（2）测试工具本身提供API接口，测试人员可以通过编写程序调用该API接口进行提交，（3）测试人员将URL集合按照一定的格式写入到文件中，然后上传到测试工具的服务器上，使得后者可以读取。测试工具需要提供“检测风险项”的选择列表，测试人员可根据测试计划选择不同的风险检测项。测试工具在测试过程中，应当对访问目标网站的速度进行控制，保证目标网站不会因为同一时刻的请求数过高，导致网站响应变慢或崩溃。测试人员在设定测试任务的基本信息时，应当根据目标网站的性能情况填入“每秒请求数”的最大值。测试工具在测试过程中应当保证每秒

发送请求的总数不超过该数值。3）测试报告，在安全测试各步骤都完成后，输出测试报告。测试报告一般包含总览页面，内容包括：（1）根据测试过程产生的各种数据，输出目标网站安全性的概要性结论；（2）测试过程发现的总漏洞数，以及按照不同安全等级维度进行统计的漏洞数据。测试报告应详细列出每一项风险的详细信息。详细信息包括：风险名称、风险等级、修复方案等关于风险的基本信息，证明风险存在的证据，包括复现风险情况的步骤和方法，测试过程中被用于证实存在风险的原始请求数据包和网站响应数据包，以及在原始请求数据包中指出触发漏洞的关键点。

目前动态应用程序安全测试（DAST）主要厂商及工具包括国外的Micro Focus Fortify WebInspect、Veracode 等，以及国内的各类安全漏洞扫描工具等。

5. 实时应用自我保护

实时应用自我保护（RASP）是一种运行时应用自我保护程序，可自身注入到应用程序中，与应用程序融为一体，实时监测、阻断攻击，使程序自身拥有自保护的能力，并且应用程序无需在编码时进行任何的修改，只需进行简单的配置即可。通过 RASP 可以实现对关键函数的监控，获取关键函数的参数信息（如对数据库操作进行监控）。

通过 RASP 的基本原理是注入到被保护的应用中，替换关键函数，获取到应用运行时的上下文，根据运行时上下文或者敏感操作，对攻击进行精准的识别或拦截。实时采集 Web 应用的高风险行为，在安全测试阶段可以辅助测试人员提前发现安全漏洞，在业务线上运行阶段

可以实时检测到外部攻击和漏洞利用，可检测的风险包括 SQL 注入、命令注入、代码执行、上传漏洞、文件读取等。同时通过特征规则、上下文语义分析及第三方安全产品数据关联分析等多种安全模型来提升检测准确率，相较于传统 Web 应用安全产品，RASP 从海量的攻击中排除掉了大量的无效攻击，聚焦发现真实的安全威胁。

目前实时应用自我保护（RASP）主要厂商以及工具包括国外的 Micro Focus、Prevoty、Waretek，以及国内的安百科技的灵犀、百度安全的 OpenRasp 等。

6. Web 应用防火墙

Web 应用防护墙（WAF）作为 Web 安全主要防护手段，是通过执行一系列针对 HTTP/HTTPS 的安全策略来专门为 Web 应用提供保护，主要用于防御针对网络应用层的攻击，如 SQL 注入、XSS 跨站脚本攻击、文件包含攻击、CC 拒绝服务攻击等。

WAF 一般部署在 web 服务器前面或者作为一个模块嵌入在 web 服务器里面，对请求的入流量和出流量均可以进行过滤检测或处理。1）请求入流量方向上，在用户请求到达应用程序前对用户请求进行检测过滤，采用基于规则的模式特征匹配或基于语义理解和机器学习的检测方案，分析并校验每个用户请求的网络包，拦截恶意攻击流量，放过正常请求，确保每个用户请求有效且安全。2）出流量响应方向上，在响应到达客户端之前对响应请求内容进行内容检测，可以准确防护恶意攻击及避免敏感信息泄漏等。同时，在处理响应时通过下发 JS

方式，结合风控大数据，解决一些风控场景需求，如防刷，打击羊毛党等。

研发运营安全体系中，在预发布阶段流水线中加入检测是否接入WAF防护的质量红线，保证业务服务上线环境处于WAF安全防护状态，在安全运营响应阶段 WAF 作为重要的安全能力工具对线上攻击请求进行实时防护。以默认安全为原则，WAF在和Web接入层直接交互结合后，业务通过Web接入层开放外网Web服务时，就默认带有WAF防护能力，可以及时有效的阻断实际产生的某些攻击尝试和行为。WAF的所有基础配置和策略下发可以由WAF运营人员统一实施，无需业务介入，就默认带有基础防护能力。同时，WAF提供防护能力开放平台，用户可以在 WAF 开放平台自助便捷操作，实现场景化定制防护需求。

目前Web应用防火墙(WAF)主要厂商包括国外的Imperva、Akamai、Cloudflare、AWS等，以及国内的阿里云、腾讯云、奇安信、绿盟等。

四、 研发运营安全发展趋势展望

随着信息化数字化的不断发展，软件应用服务自身安全重视程度也将逐步增加，研发运营安全体系将会同步完善，具体表现为：

研发运营安全管理体系将更加完善。管理制度流程是实践研发运营安全的基础，随着研发运营安全理念的不断深化，将会推动相应安全管理体系的不断完善。

研发运营安全体系将会推动安全技术、工具的进一步发展。SDL理念推出发展至今已经10余年，仅靠管理制度、安全人员实现研发运营安全难度是巨大的，安全技术、解决方案的出现将会促进研发运

营安全的实践落地。相应的，企业在实践研发运营安全的过程中，对于相关安全技术、工具也会提出新的能力要求，进一步推动安全技术的发展。

研发运营安全将增强安全可信生态布局。安全可信生态布局具体指两个方面，1）**企业合作共建安全可信生态，**软件应用服务涉及各个行业和领域，随着研发运营安全意识的不断提升，各行业领域领军企业将合作共建安全可信生态，满足不同用户、不同行业、不同场景的安全可信需求；2）**对于供应链安全要求将会越来越高，**软件应用服务涉及众多第三方开源及商用组件，供应链的安全对于软件应用服务自身安全可信至关重要，研发运营安全的持续推进将会提升整体供应链的安全性要求。

附录：研发运营安全优秀实践案例

（一）华为云可信研发运营案例

1）当前研发运营安全的痛点

云市场面临的主要威胁:数据泄露,身份、凭证和访问管理不足，不安全的接口和应用程序编程接口（API），系统漏洞，账户劫持，恶意的内部人员，高级持续性威胁（APT），数据丢失，尽职调查不足，滥用和恶意使用云服务，拒绝服务（DoS），共享的技术漏洞等；

企业上云的关键安全诉求：业务连续不中断，如防网络攻击，防黑客入侵，法律遵从、合规等；运维全程可管控，如配置安全策略，风险识别和处置，操作可审计、追溯等；数据保密不扩散，如防外部窃取，内部非授权员工不可见，云服务商不可见等。

2）研发运营安全的落地实现

华为云在研发运营生命周期中,可信是云服务最重要也是最根本的要求,建立并完善信任机制是业务第一优先级。在充分分析业界对可信的解读并广泛听取客户反馈后,华为云认为云服务的可信应该包括安全、隐私、合规、韧性、透明五个基本特征。

（1）管理制度

华为云产品团队（开发、运营、运维、安全等）作为产品可信的第一责任部门，在服务产品生命周期的各项活动中落实可信要求，构建产品的可信能力，打造“过程可信”+“结果可信”的高质量产品，同时进行自查自纠，主动发现问题并进行改进。

在可信战略和框架指引下，从公司到华为云各级部门都致力于建立可信的文化，牵引全员可信意识的转变，把可信根植于企业的各个方面。建立软件工程可信胜任的管理要求，开展对软件业务主管、Committer、软件架构师、开发工程师、和测试工程师等角色的资质管理及选拔、调整。软件工程师需要学习可信的要求并通过考试，使其具备编写可信的高质量代码的能力，以确保实现产品可信的要求。

（2）安全要求

华为云在内部 DevSecOps 实践的基础上，推出了商业化的 DevCloud 软件开发产品，除了对外提供服务，DevCloud 也是华为云研发运维团队使用的持续集成、持续交付平台，它将规范、基线、软件/API、用例，工具固化到 DevSecOps 流水线，形成自动化、可视化的服务全生命周期管理模式。

（3）安全隐私需求分析与设计

内嵌可信的开发运维流程。在过去 20 年里，IPD 研发流程帮助华为将 ICT 产品的质量提升到了新的高度；今天，为了适应云环境下快速交付服务的需求，华为云在吸收业界先进理念的基础上，持续改进开发和运维流程，形成了开发、运维、安全一体化的 DevSecOps 可信软件工程实践。

华为云的 DevSecOps 可信软件工程实践通过工具和技术规范实现了流程的固化，使过程和结果透明可见、从故障现象到模块代码可追溯，从而实现云服务全生命周期的过程可信。

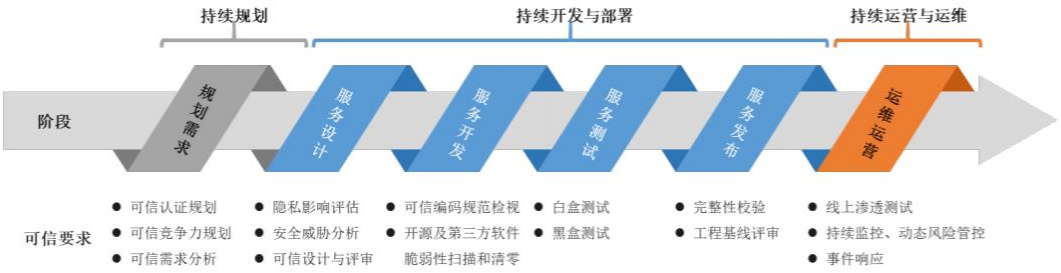


图 1 华为云 DevSecOps 生命周期模型

全生命周期的数据安全。华为云以区域为单位提供服务，区域即是客户自主选择内容数据的存储位置，华为云未经授权绝不会跨区域移动客户的内容数据；通过不同粒度的访问控制机制，确保客户只能访问到自己的数据。具体体现为 1）华为云通过数据加密服务（DEW）的专属加密（DHSM）、密钥管理（KMS）、密钥对管理等功能提供云上数据加密和存储保护；2）使用虚拟专用网络（VPN）在远端用户和 VPC 之间建立符合行业标准的安全加密通信隧道，将已有数据中心无缝扩展到华为云上，提供租户端到端的数据传输机密性保障。通过 VPN 在传统数据中心与 VPC 之间建立通信隧道，客户可方便地使用华为云的资源；3）华为云服务通过标准 RESTful 形式向外发布，全网数据传输使用 TLS 进行加密保护，同时也支持基于 X.509 证书的目标网站身份认证；4）华为云提供多重冗余和容灾机制保证数据的高持久和服务高可用，其中不少产品的可用性指标均达到业界先进水平。

（4）研发与验证

华为云开发者中心通过提供开发环境、OpenAPI 和 SDKs 以及基于生命周期的一站式应用开发服务，使软件开发更加简单高效。同时提供代码检查、测试管理、源代码控制、问题和缺陷跟踪等工具，帮

助开发者实现产品的安全可信。交付安全可信的软件产品一直是华为公司倡导的企业文化，华为云认为，云服务的可信不仅应该体现在技术和产品上，更应该根植于从需求规划、架构设计、系统开发、运维运营到客户服务的全生命周期中。无论内部业务流程还是对外客户服务，华为云完全遵从法律法规和国际标准提出的安全、合规和隐私保护基本原则，制定了超过 1000 项的可信要求，将功能与质量、安全与隐私保护融入云服务全生命周期，同时特别关注个人信息在采集、使用，保留，传输、披露和处置等处理过程中的隐私保护，确保流程透明、结构完善、控制严谨、过程可追溯。

（5）发布部署

按照华为云发布部署中规定的云服务上线场景，各云服务必须完成自检，并承诺已满足上线华为云官网的可运营要求。

（6）上线运营

安全稳定的可信运营。为保证云服务的安全稳定运行，华为云建立了可信运营中心，运维运营内容包括运维权限管理、系统日志与审计、漏洞与补丁管理、事件管理、业务连续性管理等，覆盖了安全运营的事前防范，事中响应、事后审计的全生命周期。华为云在保证常规安全运营的基础上，还特别提取了合规、透明和隐私保护等可信要求融入云服务运营活动中。如监控与日志管理方面，日志保存超过 180 天满足监管合规要求，日志不保存用户个人敏感信息以符合隐私要求；漏洞和事件信息及时通知客户符合透明要求，用户可自主选择通知推送的方式满足隐私的要求。

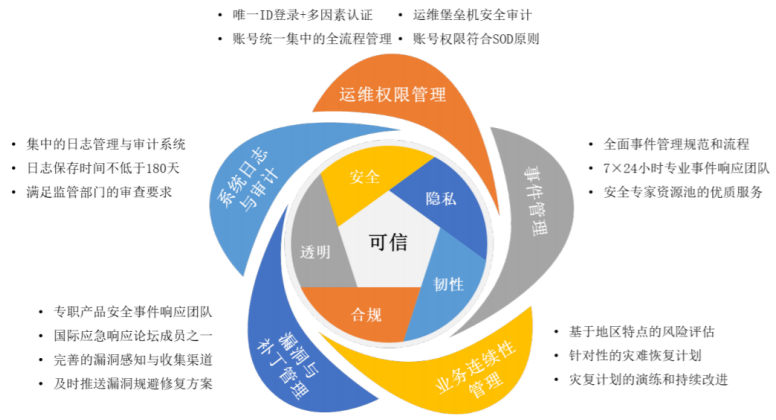


图 2 华为云可信运营

安全可信的客户服务。客户服务是云服务供应商与客户沟通的窗口和渠道，优质的服务也是可信的一种体现。华为云认为，除了完整可信的内部控制流程，透明完善的客户服务机制，也是建立信任的重要体现。华为云建立了完善透明的客户服务体系，真诚与客户建立联系，获取客户心声，收集客户需求，解决客户问题，提升产品和服务的可信能力。

华为云官网发布了清晰透明的《华为云用户协议》、《云服务等级协议（SLA）》和《隐私政策声明》，帮助客户了解华为云的责任、产品服务的指标以及隐私保护的信息；同时提供多种交互渠道，以便客户获取并行使数据主体的权利。

华为云秉承客户至上，服务第一的原则，根据基础级、开发者级、商业级、企业级不同级别的需求，建立可供选择的服务包，用户可通过在线工单、智能客服、自助服务、热线电话等多种方式获取专业的服务和帮助。

在处理服务请求时，所有涉及操作客户网络的活动必须事先获取客户的授权，并严格按照授权的范围、期限和用途进行操作，确保授

权和操作记录可追溯。同时，通过访问控制、加密、脱敏显示等技术手段，有效保护客户隐私数据。

任何用户均可通过多种渠道进行服务咨询、意见反馈和投诉建议，除基础性的站内在线客服和投诉建议热线电话外，系统复杂的企业客户可以选择适用的支持计划，获取由 IM 企业群、技术服务经理(TAM)、服务经理等组成的专属支持。

（7）停用下线

当客户决定不再使用云服务或主动删除数据时，华为云会综合内存清零、逻辑删除、虚拟卷清零、销毁加密密钥等手段确保数据及其所有副本被安全销毁。当客户注销华为云账户后，内容数据将进入保留期，客户不能访问及使用云服务，保留期届满后，内容数据会得到彻底的清除。在物理存储介质报废阶段，华为云通过对存储介质消磁、折弯或破碎等方式进行数据清除，确保其上的数据无法恢复。

3）最终效果描述

华为云在保证自身产品可信的基础上，通过全栈的服务和解决方案，把华为云积累的可信能力释放出来，帮助客户实现可信，即：安全、隐私、韧性、合规、透明五大特征。

华为云提供安全可靠、性能稳定的六大类上百种服务，并根据行业发展和客户需求不断地丰富和完善。华为云的全栈服务分类及产品示意如下图所示。

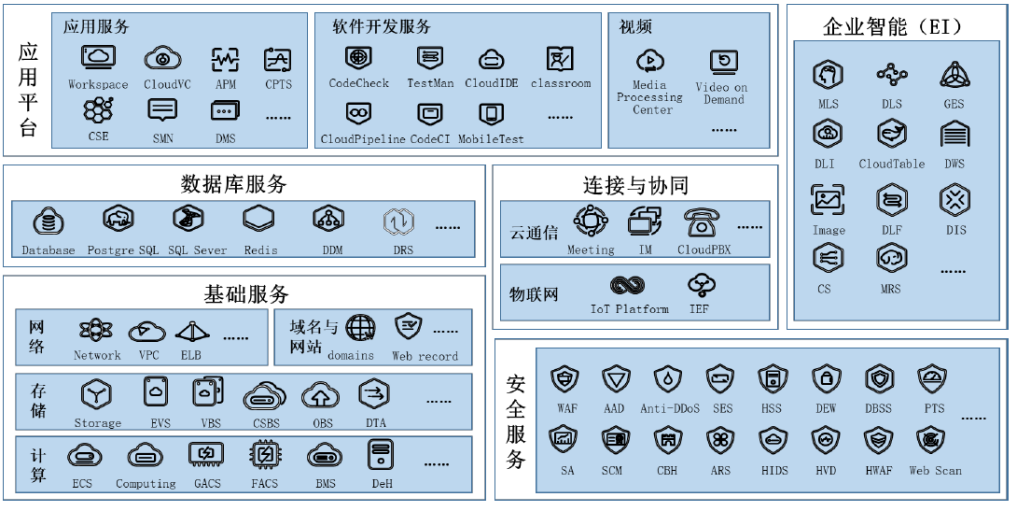


图 3 华为云全栈式服务

（二）腾讯研发运营安全实践

1）当前研发运营安全的痛点

腾讯产品研发在行业内具有一定的特色，这么多年的发展过程中，腾讯自研业务涉猎广泛几乎覆盖到了互联网所有的业务形态，重隐私要求高性能的通讯软件 IM、快速试错的各种 SNS 功能、保守的支付金融等都融合到一家公司里，这使得作为支撑业务发展的技术呈现出了巨大的复杂性和挑战。

首先的挑战，就是在研发安全运营落地实践过程中，所遇到的人力和资源有限的问题，为满足业务复杂的安全需求，就需要投入大量的资源保障研发代码、研发流程等方面的安全，而过去包括需求评审、威胁建模、安全开发、安全扫描等在内的安全环节的落地，都依赖于大量安全人力等资源的投入，而业务复杂性越来越多，版本发布也越来越多，就对安全工具链、自动化安全运营等方案，提出了更高的要求；其次是迫切需要改变被动的安全应急的现状，业务的蓬勃发展，业务的大量上线，也同时带来了很多安全应急的事件，如何将安全工作左移，将频繁的安全应急，转变为事前安全风险的规避以及提前发现，成为迫切需要解决的问题，这就需要提升整体业务团队的安全意识，并提供可信赖的安全风险发现的解决方案或者工具；最后，就是安全与研发流程的紧密结合，无论是源代码安全扫描、黑盒安全扫描、高危组件安全扫描、主机安全扫描、敏感信息扫描等等，都需要与研发流程紧密结合，才能够满足快速迭代的业务需求，而这也是早期安全面临的安全挑战。

正是如此，腾讯内部研发安全运营，也是伴随着 DevOps 等研发理念的产生和发展，越来越多的业务开始尝试各类新型的研发模式的现状下，在早期就展开了研发运营安全的探索和实践，旨在解决所遇到的与已有的安全保障体系结合的困难和挑战。

2）研发运营安全的落地实现

（1）建立管理制度

公司安全规范，为便于员工及时了解安全需求，公司发布包括《研发安全规范》、《运维安全规范》等在内的多项安全标准，并定期更新，保持安全合规以及安全机制与时俱进；各集团事业群也会结合自身业务特点不断细化，制定适合自己形态的安全规范、检查表等；配备专职安全团队，进行安全评估，对于重点业务、重大的风险进行专项跟进。

安全培训，腾讯学院专栏安全课程，包括网课以及线下培训课程，所有入职新人都必须参加相应的培训课程，其中包括安全课程；同时各集团事业群或者部门也会定期组织安全培训和考试。

（2）落实安全要求

腾讯内部从集团层面提出业务上线的安全要求，全面覆盖安全分级管理、权限管理、源码安全管理、系统上线流程、权限申请流程以及相关的算法标准、口令标准、传输安全标准等基础安全要求，与此同时，提出可落地的运维安全基线要求、研发安全基线要求，确保业务系统在设计、开发、运维等阶段需满足的安全要求，旨在收敛内网安全风险，保障数据安全及业务安全稳定运行，例如在安全编码领域，

集团发布编码安全规范，覆盖多种常规语言的编码规范，深度赋能开发人员安全实践。

（3）安全隐私需求分析与设计

腾讯 PBD 是具有腾讯特色的隐私保护方法论，即用户（Person）、控制（Button）和数据（Data），以彰显大数据时代腾讯隐私增强用户数据控制力，尊重用户隐私体验价值的努力。

图 1 腾讯 PBD 体系

用户（Person）是腾讯隐私的出发点和最终目标。在产品设计中体现为尊重用户，努力提升用户对隐私保护的感知度与参与度，最大程度地避免产品设计中的隐私缺陷。让用户在使用产品的每个环节都清晰地了解到其可能被收集的个人信息及具体使用情况，给予用户充分的自主决定权，排除用户在隐私泄露方面的担忧，使其获得最佳的用户体验。

控制（Button）致力于实现用户对个人信息的有效掌控与自主决定，增加用户操作的便捷性。将用户的数据控制力转化为一个个按键，

在收集个人信息的具体环节中通过弹窗提示、再次授权等技术手段，充分实现可视化操作，使隐私真正为用户所掌握。

数据（Data）重点关注用户个人信息的开发利用与隐私保护之间的平衡。个人信息的开发利用是为了让用户享受更加方便快捷的社交、娱乐、生活等体验。腾讯在使用个人信息以提供更好服务的同时，亦充分考虑隐私保护，避免过度收集个人信息和超越目的范围使用的行为。

与此同时，腾讯安全研发也一直贯彻“通过设计保护隐私（Privacy by Design）”，这也体现了技术、法律和价值的融合。该理念包含七大原则，指向产品设计、用户体验与价值功能三个层面。

图 2 通过设计保护隐私

（4）研发与验证安全建设

此阶段，在腾讯内部，主要涉及应用漏洞扫描，通过“应用漏洞扫描”的内部开源系统项目，在腾讯多年建设的已有的研发漏洞发现能力上，集合公司各安全团队的力量，以内部开源协同方式，打造更强更好用的公司级的安全工具，包括且不限于以下：1）SAST，静态代码扫描“啄木鸟”，开发人员可以直接在 CI 平台上使用相应插件进行自动化的扫描以及结果的查看和疑似风险的处理；另外针对内部

开源代码建设了一套自动化的机制进行代码安全的检查并且有相应的打分/激励以及推动机制逐步消除风险项；代码安全扫描结果以及风险情况也会同步显示在内部代码托管平台上，方便开发者随时查看和处理。2）DAST，Web 漏洞审计“洞犀”/App 漏洞审计“金刚”，针对业界常见以及腾讯遇到过的漏洞类型，持续打造建设相适应的 Web 和 App 漏洞审计平台；对于 Web 类型漏洞的扫描，相比业界专门优化了很多漏洞的识别算法，支持根据业务容量的限速，特别完善了测试用例无害化，使安全测试对业务的影响降到最低；对于 App 漏洞，除了传统的漏洞以外，借助于模拟器集群、UI 测试覆盖、污点追踪等技术，对于一些需要前置条件的漏洞以及隐私合规方面的深层次安全风险进行自动化的识别。3）IAST“洞犀”，结合流量代理技术，建立自动化的上线前安全扫描机制。研发人员发布之后，测试人员只需要关注功能测试，整个系统自动抓取期间的 CGI 流量并且提交“洞犀”进行安全扫描，有问题通过安全工单跟踪；结合 RASP 和 DAST 技术，可以全自动的采集流量触发 DAST“洞犀”扫描，结合 IAST 技术发现一些以往单纯靠 DAST 无法发现的漏洞点，确保覆盖的情况下扫描速度提升巨大，并且对于一些开发人员不容易定位问题点的漏洞如命令注入等，可以直接给出详细漏洞触发点信息，提升开发人员效能。4）APP 加固，有针对性行的对移动应用和 apk 进行安全加固。5)Fuzzing，逐步完善 DevSecOps 下的 Fuzzing 技术和平台。6）混沌工程，建立专注于系统可用性的混沌工程平台，并不断尝试安全领域的此类探索。

（5）平台支撑安全发布部署

安全发布部署是业务上线前的重要保障，此阶段主要涉及以下安全措施和平台保障，包括且不限于：后台服务发布，安全加固的统一服务发布平台；APP 发布，证书和软件签名/发布；镜像安全扫描；安全加固的腾讯 tlinux 内核。

（6）上线运营安全保障

业务上线后，腾讯通过自动化的安全监控、安全运营以及及时的应急响应，为业务提供安全保障，包括且不限于：1）安全可追溯的运维访问，铁将军（服务器权限管控）：面向公司业务提供服务器实名登录、权限管理、访问控制、实名审计等安全服务，安装铁将军后可实现服务器访问统一管理，权限最优化配置；2）零信任实践，全面应用于腾讯企业内部；3）网络流量安全分析，诸如管理后台、敏感信息等的漏洞风险，同时也在入侵检测和安全攻防上得以探索与内部落地实践；4）洞犀-上线后安全扫描，高危服务：针对内外网上的高危服务（如可被直接入侵、数据泄露等）进行全端口的持续监测，对于其中可蠕虫可入侵类的风险提供发现能力。部分机房的外网开放风险可“一键防护”，帮助运维同学快速的临时止损。Web 漏洞：为了防止一些不进行上学期安全扫描的漏网之鱼，洞犀系统会对外网所有的 Web 请求进行自动化的采集、去脏、去重等处理，识别出有效的 CGI 及参数并进行安全扫描；5）金刚-上线后安全扫描，面向公司 APP 产品提供全方位通用漏洞审计功能，在产品发布前，通过对 APP 的安全做全面的自动化检查，最大限度帮助业务解决 APP 中存在的安全漏洞问题；6）洋葱（入侵检测系统），基于腾讯十余年来数百万台生产

业务系统服务器的安全防护经验，提供全面、可靠的服务器安全解决方案，方案由轻量级 Agent 探针和管理服务平台组成，支持统一的资产管理、入侵检测、安全审计、安全漏洞、安全基线等；7）RASP 方案 TRASP，通过分析应用的运行行为以及上下文来发现 Web 应用安全威胁，并精准定位到漏洞源，由于与应用融为一体，可实时监测、阻断攻击，使应用自身拥有自保护的能力，可以实时检测发现针对 web 系统的各类入侵行为，包括 SQL 注入、命令注入、任意文件上传、任意文件读取、代码执行等；8）腾讯安全应急响应中心（TSRC）漏洞奖励计划，提供资金激励吸引外部安全研究人员帮助腾讯发现更多潜在安全风险和漏洞，提供其他公司快速搭建 SRC 平台的开源版本以及 SaaS 版本，目前已经有 10+知名公司使用；9）腾讯蓝军渗透测试，十余年来专注于前沿安全攻防技术研究、腾讯网络安全实战演练、腾讯业务系统安全评估等方面，站在 APT 黑客的视角去模拟攻击，全方位检验安全防护策略、响应机制的充分性与有效性，最大限度发现业务系统的潜在风险，提出解决方案及协助改进；10）安全事件应急响应小组，7*24 小时安全事件应急响应，第一时间响应，降低或消除事件影响；11）宙斯盾（DDoS 防护），通过 IP 画像、行为模式分析、Cookie 挑战等多维算法，并结合 AI 智能引擎持续更新防护策略，能够有效防御各类型 DDoS 和 CC 等攻击行为；12）门神（WAF），针对 Web 攻击进行防御，丰富的攻击特征库，配合语义分析+AI 智能检测引擎提高准确率降低误报，提供自定义打击策略以及防 CC 恶意数据爬取阻断等能力；13）安全服务中心；14）安全事件工单系统；15）

威胁情报，TSRC 安全情报平台，主动及时感知外部安全情报，提供100+知名软件的官方更新情报，5 分钟内发现，30+外部公司使用。

3）最终效果描述

整个信息产业发展到今天，软件系统日益庞大复杂，从业人员日益增多。单纯的依靠安全岗位这一角色来保证研发的系统不出现安全问题是很不现实的，这也是为何业界不断的爆出黑客入侵、数据泄露等安全事件的本质原因。于当下，安全不仅仅是安全工程师的责任，而应该成为一个组织整体的最高目标之一，需要每一个工程师的参与。腾讯安全和研发团队都要参与到研发安全运营的不断建设和优化中来，不断的推进理论和工具链的向前发展，不断的提供更好更便利的安全能力并且尽可能的通过自动化、工具化等方式为研发工程师赋能，使大家都能够承担起应有的安全责任，才能进行更好的安全管控。

（三）国家基因库生命大数据平台研发运营安全案例

1）当前研发运营安全的痛点

随着科学研究方法和技术的快速迭代和应用，相关领域的企业、高校和医院等单位和机构产生、管理和利用的生命科学数据规模不断增加。但面对海量数据，相关的单位和机构难以为数据提供安全性和可用性较高的本地存储计算资源和对外的共享应用能力。往往由于负责数据相关工作的人力队伍建设和配套流程不完善，导致国家的遗传资源的共享和应用都存在安全问题，有很大的隐患。

另外在数据共享和应用过程中，数据所有方的权益无法得到有效保障，共享的数据的安全和相关的隐私保护难以保证，导致很多研究机构、医疗机构、公司或者其它生命数据的拥有者在数据集中保存和共享方面积极性不高。

2）研发运营安全的落地实现

（1）管理制度

数字化时代，网络安全威胁已成为生命大数据平台运营的主要挑战之一，日益严峻的安全挑战将会威胁生命大数据平台的信息安全，数据安全已成为当前最为主要的安全问题。通过建设整体的网络安全架构，按照国家信息安全等级保护认证的要求，获得国家信息安全等级保护资质的三级/四级认证。

信息安全管理主要围绕防数据泄露、防攻、防特权、合法合规四个基本方向，从建立安全组织体系，信息安全管理与策略体系，云管端纵深立体信息安全技术防护体系规划入手，以数据安全保护为抓手，

推进核心安全能力建设,实现生物信息数据防窃取、防滥用、防丢失、防篡改、防误用,保障国家生物数据安全。

建立生物信息安全管理体系

面向国家安全体系战略需求，从伦理、人遗、实验室安全、信息安全、物理安全、合规性等方面建立全方位的生物信息安全体系。

（a）生命伦理安全管理

建立伦理委员会（CNGB-IRB），设立常设办公室，配备专职秘书和顾问团队，制定伦理审核及监督管理的制度，规范涉及人、实验动物、微生物等领域的研究和应用，保证国家基因库各项工作符合生命伦理国际准则、国家法规和行业规范等。

(b)遗传资源安全管理

按照《中华人民共和国人类遗传资源管理条例》（2019 年 7 月 1 日）、《生物遗传资源获取与惠益分享管理条例（草案）》、《濒危野生动植物国际贸易公约》以及《中华人民共和国野生动植物保护法》等相关法律法规，制定遗传资源管理制度及操作流程，确保遗传资源受到管控，在操作和交流过程中符合国家相关法律、法规的要求。通过生命伦理安全管理以及遗传资源安全管理从而避免生物技术不被误用和滥用。

(c)实验室安全管理

通过实验室安全管理,避免生物危害因子对实验操作人员造成伤害以及对周围环境造成污染,从而保证科研活动的正常进行。按照国家生物安全等级进行实验室建设,保证工作人员不受病源微生物等伤

害，保证产生的废弃生物材料等通过合法合规途径处理，不会对外部环境造成不利影响。

(d)物理安全

通过建设由监控系统、安保岗亭（身份+车辆识别系统）、门禁系统、内部管理系统、核心区域授权管制系统等构成的5道物理安全防线。通过人防、技防、物防的有机配合，保障各区域的物理安全。通过警民联动构建了第六道防线。采用公安“雪亮工程”，以综治信息化为支撑、以网格化管理为基础让国家基因库的安防系统与公安系统对接，充分发挥视频监控系统的作用，推进国家基因库治安防控体系建设。

（2）安全要求

为遵循《中华人民共和国人类遗传资源管理条例》，面向生物技术和生物大数据科学前沿，面向国家生物安全体系战略需求，以及我国生物健康，生物医药，生物农业等产业发展需求，我国精准医疗，数字化地球等重大科技公关等科学研究的重大需求，提升生命科学数据的安全有效共享和应用，国家基因库构建生命大数据平台(CNGBdb)，以确保中华人民共和国遗传资源得到充分保护的前提下，为国内相关机构和单位提供安全的数据共享环境，促进生命科学数据开放共享，为科学研究和生物技术产业化过程中的数据共享及应用提供保障。

（3）安全隐私需求分析与设计

1）建立安全可靠的数据汇交系统，针对我国科学研究和产业发展的生物样本资源，数据资源等多组学信息数据资源，为科研机构、

医药和企业的样本和测序数据提供统一的管理和共享，参考国际和国内的组学数据标准和管理方式，建立安全可靠的数据汇交系统。

2）建设存读一体化生物数据安全和高效传输系统，依托深圳国家基因库（以下简称：国家基因库）样本库、读平台和信息库组织结构的优势，样本库丰富的样本资源，通过就地对生物样本库的样本进行数据采集和数字化，形成统一格式的数据，并就地存入生物信息数据库，并进行生物信息学分析和挖掘，实现生物样本资源库和生物信息数据库进行有机地结合，建立生命数据全周期管理和回溯的基石。

3）基于区块链和多方安全计算技术实现可溯源、可监管的数据共享应用区块链技术，建立覆盖数据计算的生命科学数据关键节点信息上链，保证各模块及处理步骤都是可预测，可追溯和可验证，可监管，最大程度地保护数据的安全。配合区块链技术，通过安全多方计算实现数据联合分析，数据“虽彼此不可见，但可共享使用”（即“可用不可见”）的方式，与其他科研人员协同分析。

4）基于中国个人信息相关法律法规制定平台隐私政策，CNGBdb 作为数据服务的提供者给研究人员提供数据归档和管理等服务，CNGBdb 平台隐私政策遵循中国个人信息安全相关的法律法规。

（4）研发与验证

CNGBdb 建立了安全可靠的序列归档系统（CNSA），CNSA 是一个方便快捷的科学研究项目、样本、实验、测序、组装、变异等生命科学数据汇交和共享系统，结合国际核酸序列数据库联盟（International Nucleotide Sequence Database Collaboration,

INSDC）标准和 DataCite 标准，建立了数据归档存储和管理的标准规范，并接受来自全球科研的测序数据、信息和其他分析结果数据的递交，为全球的研究者提供当前最全面的数据和信息资源，使研究人员能够更便捷地访问数据，促进数据的再利用。

（5）发布部署

基于云平台通过可信云服务认证，CNGBdb 通过 ISO27001 信息安全管理体系认证，从服务协议标准性、数据存储可靠性、用户数据私密性、业务可用性、功能完备性、运维系统完善性等多方面达到国内顶级云服务评测系统的认证标准，对外发布 CNGBdb 平台业务并提供公益性服务，保障信息资源的安全，保护信息化进程健康、有序、可持续发展。

（6）上线运营

(a)基础安全

具备主动发现安全漏洞和提供修复方案以及漏洞修复能力；具备对网络攻击和明显异常行为的主动感知和防御能力；对安全事件应急响应和回溯能力。

(b)业务安全

主动发现潜在业务安全风险，提供解决方案和风险控制在可接受范围内；有能力支撑和应对业务正常发展和运营过程中以及业务场景变化可能带来新的业务安全风险。

(c)数据安全

主动识别数据在全生命周期内和流动过程中的数据安全风险和评估出合理风险等级，并将风险控制在可接受范围内；具备对数据泄露事件的应急响应和追溯调查能力。

(d)安全合规

满足安全监管、法律要求，有效支持基因库业务开展、运营和外部合作；满足合规的基础上，构建符合基因库自身特性的安全合规体系。

（7）服务停用下线

因用户申请或业务需要，对平台上的特定数据进行彻底删除的操作，在进行数据的销毁处理时,需首要遵循知情同意、隐私和保密的原则。在数据销毁执行环节，依据 DOD 5220.22-M 标准，保证磁盘中存储数据的永久删除，不可恢复。若销毁对象被其他对象所关联，则将该对象与关联对象一并销毁/删除；若销毁对象未被其他对象所关联，则只销毁该对象本身。

3）最终效果描述

国家基因库生命大数据平台（CNGBdb）面向国家安全体系战略需求，从安全汇交、安全传输、安全管理和安全共享等方面建立了数据共享安全运营体系，在民生健康、重大疫情防控、分子育种、全球物种多样性保护等方面建立了多组学数据资源管理和应用体系，为抢占未来生命科学科技创新战略制高点提供了重要支撑和基础条件，也为推动我国生物数据安全和维护数据主权提供了重要保障。

截至 6 月 8 日，CNGBdb 核酸序列归档系统归档的数据总量为 2243TB，用户在 CNSA 递交的数据在被国际包含 Nature，Cell 等在内的 80 多个期刊杂志接收，论文成功发表。CNSA 汇交了来源 100 多个递交单位的生命科学数据。

CNGBdb 支撑大型国际科研合作项目，包括但不限于 ICGC-ARGO 项目，地球生物基因组计划(EBP)，万种鱼基因组项目，万种植物基因组项目，万种药用植物项目等大型国际国内合作项目，共享 CNGBdb 基础设施能力。与世界最大的药用植物园-广西药用植物园合作搭建万种药用植物数据库（10KMP），包含近 1000 种药用植物的元信息及转录组测序数据。CNGBdb 与联合国粮食及农业组织 FAO GLIS 系统的 DOI 对接，实现物种、组学等信息和数据的有效互联互通，避免信息重复，方便数据统一追踪和索引，促进科学研究。CNGBdb 与中国科学院战略生物资源服务网络、全球基因组生物多样性联盟，以及国家人类遗传资源共享服务平台等组织达成合作互通，共享资源信息。2019 年，国家基因库加入了中国科学院战略生物资源服务网络，新增资源互通信息 5838 份，同时，国家基因库参与了生物资源信息平台建设及其应用研讨会，并加入了中国科学院生物资源目录。

应用区块链和多方安全技术，CNGBdb 发布国内首个基于区块链和安全多方计算的新冠病毒基因组分析平台。该分析平台展示了现有公开新冠病毒数据集（来自 GISAID、NCBI、CNGBdb 等）的演化树分析结果，包括样本序列演化关系、地理位置、采样时间等，可实时追踪病毒流行病学情况、预测未来毒株演化。此外，还能帮助用户大大

节约维护、更新数据集的时间成本。基于区块链和安全多方计算技术搭建多方安全计算工具，允许用户在不公布己方数据的前提下，联合其他科研人员协同分析并共享结果，同时结合区块链技术，保证所有数据和计算过程均可回溯且不可篡改。其正式上线意味着生命科学大数据的安全共享和开发利用上了一个新台阶。