奇安信

北京2022年冬奥会官方赞助商
Official Sponsor of the Olympic Winter Games Beijing 2022

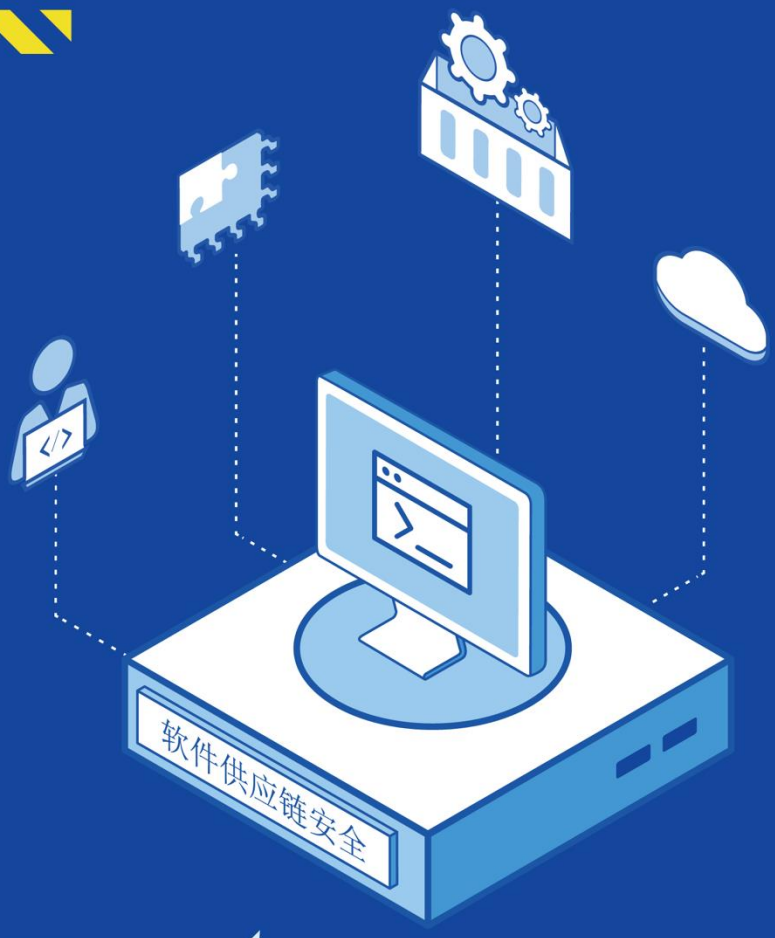
https://www.qianxin.com

2021 中国软件供应链安全分析报告

THE REPORT

发布机构：

奇安信代码安全实验室



目　录

一、前言

数字化时代，软件无处不在。软件如同社会中的“虚拟人”，已经成为支撑社会正常运转的最基本元素之一，软件的安全性问题也正在成为当今社会的根本性、基础性问题。

随着软件产业的快速发展，软件供应链也越发复杂多元，复杂的软件供应链会引入一系列的安全问题，导致信息系统的整体安全防护难度越来越大。近年来，针对软件供应链的安全攻击事件一直呈快速增长态势，造成的危害也越来越严重。

2020 年 4 月，Rubygems 开源软件包生态系统被放入了数百个恶意软件包，这些恶意软件包的下载总量近 10 万次。例如，“atlas-client”是一个诱骗的诱饵程序包，用来仿冒“atlas_client”，被下载了超过 2100 次。

2020 年 5 月，GitHub 披露了针对 Apache NetBeans IDE 项目的开源软件供应链攻击 Octopus Scanner，最终统计显示，有 26 个开源项目被植入了 Octopus Scanner 后门。

2020 年 12 月，全球著名的网络安全管理软件供应商 SolarWinds 遭遇国家级 APT 团伙高度复杂的供应链攻击。该攻击直接导致包括美国关键基础设施、军队、政府等在内的超过 18000 家客户全部受到影响，可任由攻击者完全操控。

2021 年 2 月，安全研究人员通过利用开源生态安全机制上的漏洞，成功侵入了微软、苹果、PayPal、特斯拉、优步等 35 家国际大

型科技公司的内网，这种新颖的软件供应链攻击方式被定义为依赖混淆攻击。

2021 年 3 月，PHP 的 Git 服务器被攻击，攻击者向 git.php.net 服务器上的 php-src 存储库推送了两次恶意提交，在 PHP 代码中植入了一个后门，其目标是可以通过该后门获得运行 PHP 的网站系统的远程代码执行权限。

2021 年 4 月，知名代码测试公司 Codecov 宣布其产品的 bash uploader 脚本被攻击者修改，导致用户在使用 Codecov 产品时，会向攻击者的服务器发送敏感信息，从而导致攻击者可以获取用户的软件源代码等机密信息。

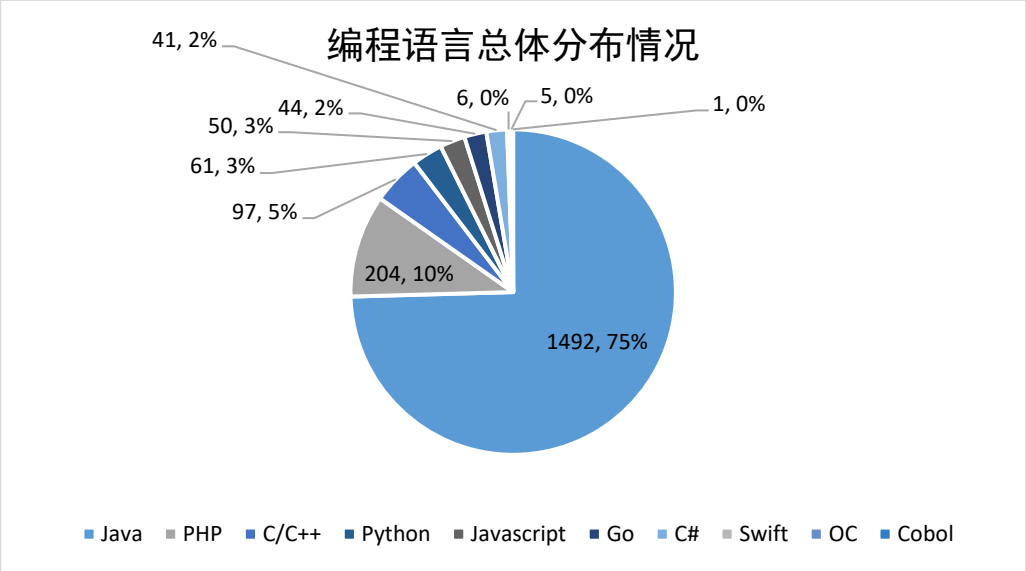
攻击不断左移，针对软件供应链的攻击事件频发，系统性的研究软件供应链安全，防范软件供应链安全风险，已经迫在眉睫。2020 年，奇安信代码安全实验室依托自身在软件安全领域十余年的技术积累，针对国内软件供应链的安全状况进行了大量的研究、实践和数据分析工作，形成本报告。报告内容主要包括国内企业自主开发源代码安全状况分析、开源软件生态发展与安全状况分析、国内企业软件开发中开源软件应用状况分析、典型应用系统供应链安全风险实例分析、总结及建议等五个方面，希望可以为相关单位开展软件供应链安全相关的研究和实践工作提供借鉴和参考。

二、国内企业自主开发源代码安全状况

源代码是软件的原始形态，位于软件供应链的源头。源代码安全是软件供应链安全的基础，其地位非常关键和重要。2020 年全年，奇安信代码安全实验室对 2001 个国内企业自主开发的软件项目源代码进行了安全缺陷检测，检测的代码总量为 335011173 行，共发现安全缺陷 3387642 个，其中高危缺陷 361812 个，整体缺陷密度为 10.11 个/千行，高危缺陷密度为 1.08 个/千行。

1、编程语言分布情况

在被检测的 2001 个国内企业自主开发的软件项目中，使用数量排名前 3 的编程语言为 Java、PHP、C/C++，对应的软件项目数量分别为 1492 个、204 个和 97 个。可以看出，相关国内企业在进行软件开发时的首选语言是 Java 语言，占比高达 75%。编程语言的总体分布情况如下图所示。



2、典型安全缺陷检出情况

输入验证、路径遍历、跨站脚本、注入、NULL 引用、资源管理、密码管理、API 误用、配置管理、日志伪造等十类安全缺陷是程序员在编写软件代码时经常会出现的典型安全缺陷。典型安全缺陷的检出率可以体现出软件源代码的基本安全状况（检出率指含有某类缺陷的软件项目数占软件项目总数的比例）。在被检测的 2001 个软件项目中，十类典型安全缺陷的总体检出率为 77.8%，每类典型缺陷的检出率及排名如下表所示。

排名	缺陷类型	检出率
1	输入验证	50.8%
2	路径遍历	39.6%
3	跨站脚本	39.5%
4	注入	37.3%
5	NULL 引用	31.8%
6	资源管理	31.6%
7	密码管理	31.0%
8	API 误用	28.7%
9	配置管理	28.0%
10	日志伪造	18.2%

三、开源软件生态发展与安全状况

Gartner 表示，现代软件大多数是被“组装”出来的，不是被“开发”出来的。据 Forrester 统计，软件开发中，80-90%的代码来自于开源软件。因此，现代软件的源代码绝大多数是混源代码，由企业自主开发的源代码和开源软件代码共同组成。开源软件是现代软件开发最基础的原材料，与企业自主开发的源代码所处的软件供应链环节相同，也位于软件供应链的源头，其代码自身的安全状况，会直接影响最终软件的安全性。

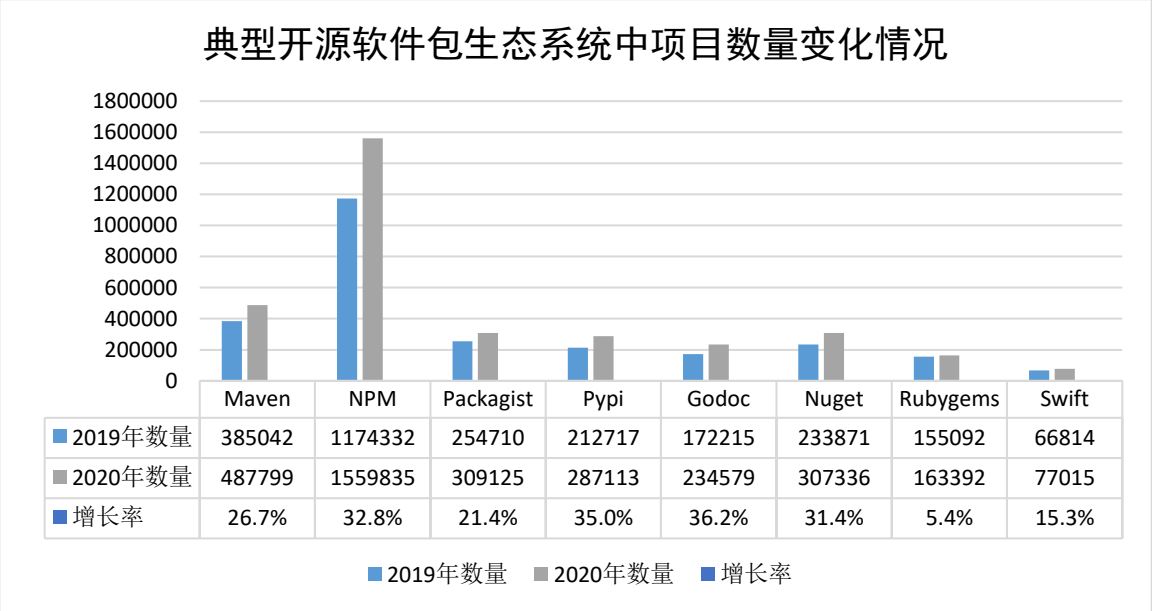
本报告从开源软件生态发展状况、开源软件源代码安全状况、开源软件公开报告漏洞状况、开源软件活跃度状况等四个方面对 2020 年开源软件生态发展与安全状况进行综合分析。

1、开源软件生态发展状况分析

据奇安信代码安全实验室监测和统计，2019 年底和 2020 年底，主流开源软件包生态系统中开源项目总量分别为 2841314 个和 3814194 个，一年间增长了 34.2%；截至 2020 年底，主流开源软件包生态系统中平均每个开源项目有 10.2 个版本。可以看出，2020 年开源软件生态更加繁荣，整体发展非常迅猛。

本报告中对八个典型的开源软件包生态系统进行了进一步的分析和比较，这八个包生态系统为 Maven、NPM、Packagist、Pypi、Godoc、Nuget、Rubygems、Swift，具体分析如下。

NPM 包生态项目数量最多，Godoc 包生态增速最快。八个典型的开源软件包生态系统中开源项目数量和增长率情况如下图所示，其中开源项目数量最多的是 NPM 包生态系统，截至 2020 年底，其开源项目数量达到了 1559835 个；开源项目数量增速最快的是 Godoc 包生态系统，2020 年一年间的项目总量增速达到了 36.2%。



	Maven	NPM	Packagist	Pypi	Godoc	Nuget	Rubygems	Swift
2019年数量	385042	1174332	254710	212717	172215	233871	155092	66814
2020年数量	487799	1559835	309125	287113	234579	307336	163392	77015
增长率	26.7%	32.8%	21.4%	35.0%	36.2%	31.4%	5.4%	15.3%

Maven、Nuget、NPM 包生态系统的开源项目开发者比较“勤奋”，开源项目的平均版本数超过 11 个。截至 2020 年底，八个典型的开源软件包生态系统的开源项目数量和版本数量如下表所示。其中，Maven 包生态系统平均每个开源项目有 18.0 个版本，Nuget 包生态系统平均每个开源项目有 11.7 个版本，NPM 包生态系统平均每个开源项目有 11.0 个版本。

序号	包生态系统	2020 年项目数	2020 年版本数	平均版本数
1	Maven	487799	8785416	18.0
2	NPM	1559835	17148119	11.0
3	Packagist	309125	3035815	9.8

4	Pypi	287113	2419533	8.4
5	Godoc	234579	1109833	4.7
6	Nuget	307336	3588880	11.7
7	Rubygems	163392	1094135	6.7
8	Swift	77015	474314	6.2

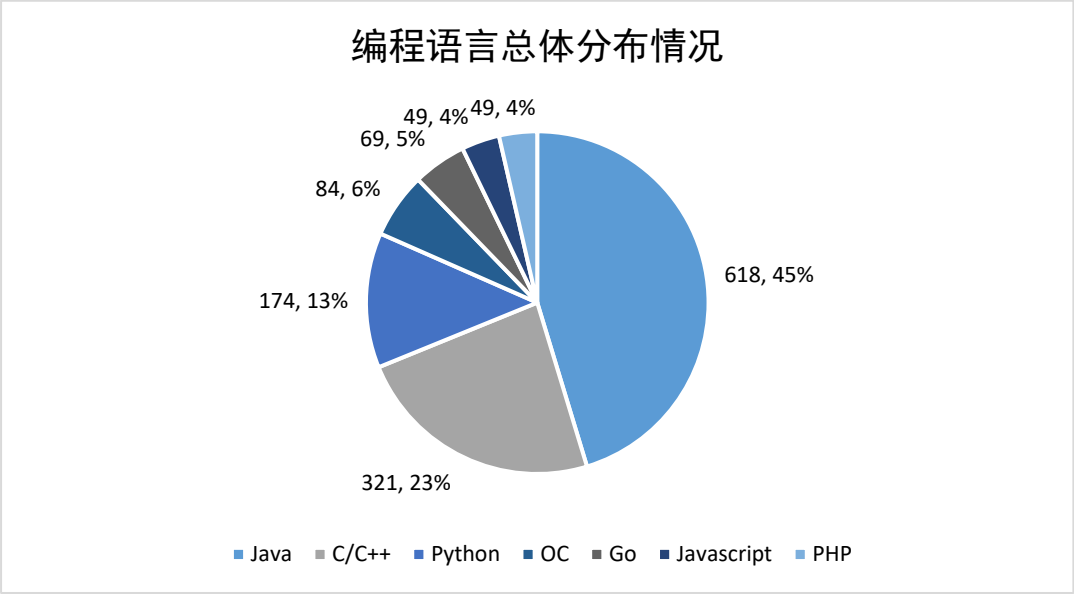
2、开源软件源代码安全状况分析

奇安信代码安全实验室于 2015 年初发起了“奇安信开源项目检测计划”，该计划是一项针对开源软件项目的公益性安全检测计划，旨在让广大开发者关注和了解开源软件的安全问题，提高软件安全开发意识和技能。

2020 年全年，“奇安信开源项目检测计划”对 1364 个开源软件项目的源代码进行了安全检测，代码总量为 124296804 行，共发现安全缺陷 1859129 个，其中高危缺陷 117738 个。2020 年检测的 1364 个开源软件项目整体缺陷密度为 14.96 个/千行，高危缺陷密度为 0.95 个/千行。

（1）编程语言分布情况

2020 年检测的 1364 个开源项目中，一共涉及到 7 种编程语言，分别是 Java、C/C++、Python、OC、Go、JavaScript、PHP，编程语言的分布情况如下图所示。



（2）典型安全缺陷检出情况

输入验证、路径遍历、跨站脚本、注入、NULL 引用、资源管理、密码管理、API 误用、配置管理、日志伪造等十类安全缺陷是程序员在编写软件代码时经常会出现的典型安全缺陷。典型安全缺陷的检出率可以体现出软件源代码的基本安全状况（检出率指含有某类缺陷的软件项目数占软件项目总数的比例）。在 2020 年检测的 1364 个开源软件项目中，十类典型安全缺陷的总体检出率为 56.3%，每类典型缺陷的检出率及排名如下表所示。

排名	缺陷类型	检出率
1	输入验证	34.9%
2	路径遍历	30.7%
3	注入	28.6%
4	NULL 引用	24.8%
5	API 误用	24.3%
6	资源管理	20.7%

7	跨站脚本	19.1%
8	日志伪造	17.9%
9	密码管理	13.8%
10	配置管理	12.9%

3、开源软件公开报告漏洞状况分析

据奇安信代码安全实验室监测与统计，截至 2020 年底，CVE/NVD、CNNVD、CNVD 等公开漏洞库中共收录开源软件相关漏洞 41342 个，其中 5366 个为 2020 年度新增漏洞。

（1）大型开源项目漏洞总数及年度增长 TOP20

截至 2020 年底，历史漏洞总数排名前 20 的大型开源项目信息如下表所示。

序号	大型开源项目	主页地址	历史漏洞总数
1	Linux Kernel	https://www.kernel.org/	4193
2	Chromium (Google Chrome)	http://www.chromium.org/Home	2658
3	Mozilla Firefox	https://www.mozilla.org/en-US/firefox/	2093
4	Thunderbird	https://www.thunderbird.net/zh-CN/	1023
5	MySQL	https://www.mysql.com/	997
6	PHP	https://www.php.net/	653
7	Wireshark	https://www.wireshark.org/	597
8	ImageMagick	https://imagemagick.org/	597
9	WordPress	https://wordpress.org/	561
10	GitLab	https://about.gitlab.com/	465
11	Moodle	https://moodle.org/	412

12	Xen Project (Hypervisor)	https://xenproject.org/	355
13	QEMU	http://www.qemu.org/	333
14	FFmpeg	https://ffmpeg.org/	324
15	Chakra Core	https://github.com/chakra-core/chakracore	279
16	phpMyAdmin	https://www.phpmyadmin.net/	265
17	Oracle VM VirtualBox	https://www.virtualbox.org/	262
18	MediaWiki	https://www.mediawiki.org/	231
19	WebKitGTK	http://webkitgtk.org/	221
20	Magento	http://www.magento.com/	188

2020 年一年间，公开报告漏洞数量增长排名前 20 的大型开源项目信息如下表所示。

序号	大型开源项目	主页地址	2020 年漏洞增量
1	Chromium (Google Chrome)	http://www.chromium.org/Home	261
2	GitLab	https://about.gitlab.com/	237
3	Linux Kernel	https://www.kernel.org/	235
4	Mozilla Firefox	https://www.mozilla.org/en-US/firefox/	157
5	Mattermost	http://www.mattermost.org/	143
6	MySQL	https://www.mysql.com/	139
7	Thunderbird	https://www.thunderbird.net/zh-CN/	81
8	Oracle VM VirtualBox	https://www.virtualbox.org/	70
9	QEMU	http://www.qemu.org/	46
10	Xen Project (Hypervisor)	https://xenproject.org/	43
11	ImageMagick	https://imagemagick.org/	43
12	FreeRDP	https://www.freerdp.com/	40
13	Magento	https://magento.com/	40

14	OpenJDK	http://openjdk.java.net/	39
15	Nextcloud	https://nextcloud.com/	37
16	TensorFlow	https://www.tensorflow.org/	35
17	MediaWiki	https://www.mediawiki.org/	35
18	Chakra Core	https://github.com/chakra-core/chakracore	32
19	Ghostscript	https://www.ghostscript.com/	27
20	WordPress	https://wordpress.org/	24

（2）主流开源软件包生态系统漏洞总数及年度增长 TOP20

截至 2020 年底，主流开源软件包生态系统中历史漏洞总数排名前 20 的开源软件信息如下表所示。

序号	开源软件	所属包生态系统	历史漏洞总数
1	TYPO3 CMS	Packagist	101
2	Ruby on Rails	Rubygems	98
3	OpenSSL	Swift	95
4	Exiv2	Conan	80
5	Plone	Pypi	79
6	Apache Struts	Maven	78
7	Django	Pypi	76
8	Dolibarr ERP & CRM	Packagist	73
9	Symfony	Packagist	73
10	Puppet	Rubygems	70
11	Drupal (core)	Packagist	65
12	silverstripe-framework	Packagist	62
13	Jackson-databind	Maven	55
14	Ansible	Pypi	53
15	keycloak	Maven	53

16	OpenShift Origin	Godoc	45
17	SQLite	Swift	42
18	Zend Framework	Packagist	39
19	libxml2	Conan	36
20	Kubernetes	Godoc	35

2020 年一年间，主流开源软件包生态系统中公开报告漏洞数量增长排名前 20 的开源软件信息如下表所示。

序号	开源软件	所属包生态系统	2020 年漏洞增量
1	Jackson-databind	Maven	26
2	keycloak	Maven	22
3	Dolibarr ERP & CRM	Packagist	20
4	Ansible	Pypi	18
5	Openfire	Maven	16
6	Electron - Cross-platform desktop application shell	NPM	16
7	Centreon	Packagist	14
8	Drupal (core)	Packagist	14
9	TYPO3 CMS	Packagist	13
10	October CMS	Packagist	13
11	Kubernetes	Godoc	12
12	SQLite	Swift	12
13	Airflow	Pypi	12
14	Vault by HashiCorp	Godoc	11
15	Apache Tomcat	Maven	10
16	Infinispan	Maven	10
17	Python Pillow	Pypi	10
18	Consul	Godoc	9
19	Plone	Pypi	9
20	Subrion	Packagist	9

4、开源软件活跃度状况分析

活跃度也是衡量开源软件安全性的一个重要维度。不活跃的开源软件，无论是更新频率很低，或者被废弃，一旦出现安全漏洞，难以得到及时的修复，安全风险很高；活跃的开源软件中，如果其版本更新发布的频率过高，同样会增加使用者运维的成本和安全风险。在选择使用开源软件时，应该充分考虑这两个因素。本报告中分析了 2020 年主流开源软件包生态系统中开源软件的版本更新情况，可以一定程度上体现当前开源软件活跃度的整体状况。

（1）61.6%的开源软件项目处于不活跃状态

我们将一年内未更新发布过版本的开源软件项目定义为不活跃项目。2020 年全年，主流开源软件包生态系统中不活跃的开源软件项目数量为 2347794 个，占比达到 61.6%。

本报告中对八个典型的开源软件包生态系统进行了进一步的分析和比较，这八个包生态系统为 Maven、NPM、Packagist、Pypi、Godoc、Nuget、Rubygems、Swift，其中 NPM 的不活跃项目数量最多，达到 1018533 个，Rubygems 的不活跃项目比例最高，占比达到 86.5%，具体数据见下表。

序号	包生态系统	项目总数	不活跃项目数	不活跃项目比例
1	Maven	487799	272555	55.9%
2	NPM	1559835	1018533	65.3%
3	Packagist	309125	208890	67.6%
4	Pypi	287113	170044	59.2%

5	Godoc	234579	143685	61.3%
6	Nuget	307336	187238	61.0%
7	Rubygems	163392	141259	86.5%
8	Swift	77015	59521	77.3%

（2）13000 多个开源软件一年内更新发布超过 100 个版本

2020 年全年，主流开源软件包生态系统中，更新发布 100 个以上版本的开源项目有 13411 个。前述八个典型的开源软件包生态系统中，一年内更新发布超过 100 个版本的项目数量见下表。

排名	包生态系统	对应的开发语言	一年内发布超过 100 个版本的项目数量
1	NPM	Javascript	8317
2	Maven	Java	2264
3	Nuget	.NET	1104
4	Pypi	Python	613
5	Packagist	PHP	554
6	Swift	Swift	281
7	Godoc	Go	193
8	Rubygems	Ruby	38

四、国内企业软件开发中开源软件应用状况

如前所述，现代软件的源代码绝大多数是混源代码，由企业自主开发的源代码和开源软件代码共同组成。本章内容将针对国内企业在进行软件开发工作时，使用开源软件的具体情况进行分析。主要回答两个问题：一是国内企业在软件开发中是否使用以及使用了多少开源软件？二是其使用的开源软件是否存在安全问题？

2020 年全年，奇安信代码安全实验室对 2557 个国内企业软件项目中使用开源软件的情况进行了分析，这些软件项目的应用领域涉及政府、金融、能源等重要行业。分析发现，国内企业在软件开发中普遍使用存在已知漏洞的开源软件，存在巨大的软件供应链安全风险。具体分析数据如下。

1、开源软件总体使用情况分析

（1）国内企业软件项目 100%使用开源软件

在被分析的 2557 个国内企业软件项目中，无一例外，均使用了开源软件。最多的项目使用了 3878 个开源软件，平均每个项目使用 126 个开源软件。使用开源软件最多的 5 个项目情况如下表所示。

项目名称	使用的开源软件数量
项目 1	3878
项目 2	3838
项目 3	3536
项目 4	3062
项目 5	2637

经过后续的调研和访谈，我们还发现，软件项目中使用的开源软件数量大大超出了软件项目管理者和程序员自身的认知。由于开源软件之间的依赖关系错综复杂，且软件开发中依赖包的管理通常通过包管理器程序自动管理，软件开发者常常意识不到自己使用了数量巨大的开源软件，因此当某个开源软件曝出安全漏洞时，软件开发者常常“躺枪”而不自知，这中间隐含了巨大的软件供应链安全风险。

（2）流行开源软件被近 1/4 的软件项目使用

一些流行开源软件会被很多软件项目所使用，这些开源软件一旦出现安全漏洞，影响面将会非常巨大。对于大型企业来说，企业内部可能就有数以百计的软件开发项目，更加需要对流行开源软件保持足够的关注和重视，应该做到对其在本单位内的使用情况心中有数。经统计，在我们分析的 2557 个国内企业软件项目中，被使用最多的开源软件为 Apache Commons Lang，被 622 个项目所使用，占比达 24.3%。被使用最多的前 5 名开源软件如下表所示。

开源软件名称	使用它的项目数量	被使用率
Apache Commons Lang	622	24.3%
Apache Commons Collections	620	24.2%
dom4j: flexible XML framework for Java	563	22.0%
Simple Logging Facade for Java (SLF4J)	510	19.9%
Javax Inject	472	18.5%

2、开源软件漏洞风险分析

（1）近 9 成软件项目存在已知开源软件漏洞

分析发现，在 2557 个国内企业软件项目中，存在已知开源软件漏洞的项目有 2280 个，占比高达 89.2%；存在已知高危开源软件漏洞的项目有 2062 个，占比为 80.6%；存在已知超危开源软件漏洞的项目有 1802 个，占比为 70.5%。

（2）平均每个软件项目存在 66 个已知开源软件漏洞

在 2557 个国内企业软件项目中，共检出 168604 个已知开源软件漏洞（涉及到 4166 个唯一 CVE 漏洞编号），平均每个软件项目存在 66 个已知开源软件漏洞，最多的软件项目存在 1200 个已知开源软件漏洞。存在已知开源软件漏洞数量排名前 5 的项目情况如下表所示。

项目	存在开源软件漏洞数量
项目 1	1200
项目 2	1013
项目 3	649
项目 4	517
项目 5	426

（3）影响最广的开源软件漏洞存在于 44.3%的软件项目中

从漏洞的影响度来分析，影响范围最大的开源软件漏洞为 CVE-2020-5421，影响了 44.3%的软件项目。影响度排名前 5 的开源软件漏洞情况如下表所示。

漏洞名称	CVE 编号	影响项目数量	影响度
Spring Framework 安全漏洞	CVE-2020-5421	1132	44.3%
Google Guava 访问控制错误漏洞	CVE-2020-8908	1021	39.9%
Apache Log4j 信任管理问题漏洞	CVE-2020-9488	1020	39.9%
FasterXML Jackson-databind 代码问题漏洞	CVE-2020-8840	905	35.4%
FasterXML Jackson-databind 代码问题漏洞	CVE-2020-25649	866	33.9%

（4）15 年前的开源软件漏洞仍然存在于多个软件项目中

分析发现，部分软件项目中存在十几年前公开的古老开源软件漏洞，最古老的漏洞是 2005 年 11 月公开的 CVE-2005-3510，仍然存在于 31 个项目中。部分古老开源软件漏洞的影响情况如下表所示。

漏洞名称	CVE 编号	发布日期	影响项目数量
Apache Tomcat 目录列表拒绝服务漏洞	CVE-2005-3510	2005.11.06	31
Jetty URL 编码的反斜杠源代码泄露漏洞	CVE-2005-3747	2005.11.22	41
Apache Tomcat 跨站脚本攻击漏洞	CVE-2005-4838	2005.12.31	32
Apache Struts ActionForm 拒绝服务漏洞	CVE-2006-1547	2006.3.30	32
Apache Struts 特定参数安全绕过漏洞	CVE-2006-1546	2006.3.30	32

3、开源软件运维风险分析

开源软件运维风险复杂多样，本报告主要从老旧开源软件的使用和开源软件多版本的使用角度进行分析。

（1）18 年前的老旧开源软件版本仍在被使用

分析发现，许多软件项目中使用了十几年前发布的开源软件版本，存在很大的运维风险。被使用的老旧开源软件版本中，最老旧的一个

是 2003 年 3 月 3 日发布的 Apache Xalan 2.5.D1，已经有 18 年之久，但仍然被 7 个软件项目所使用。按老旧程度排名前 5 的开源软件如下表所示。

开源软件名称	版本号	版本发布日期	使用它的项目数量
Apache Xalan	2.5.D1	2003.03.03	7
XML Pull Parsing API	1.1.3.1	2003.06.17	273
JDOM	1.0-FCS	2004.09.03	25
SSLExt	1.2-0	2004.10.04	17
Jboss J2se	200504122039	2005.04.26	14

（2）开源软件各版本使用非常混乱

分析发现，各个项目中开源软件使用的版本非常混乱，并非使用的都是最新版本。Spring Data 是被使用版本最多的开源软件，有 162 个版本在被使用。按照被使用版本的数量排序，排名前 5 的开源软件情况如下表所示。

开源软件名称	被使用的版本数量
Spring Data	162
Jackson-databind	141
Apache Tomcat	132
Jetty	126
Hibernate ORM	121

五、典型软件供应链安全风险实例分析

1、国内某主流 OA 系统供应链安全分析

本实例中分析的 OA 系统是一款国产 OA 系统，覆盖了流程审批、会议管理、考勤管理、假期管理等在内的各种日常办公场景，能够有效地提升组织管理与协同的效率。该系统在国内应用非常广泛，覆盖了政府、金融、医疗、建筑等数十个行业的数万个客户。

经分析发现，该 OA 系统中使用了 Spring Framework、Apache Commons FileUpload、Apache Commons Codec、Apache Log4j 等在内的超过 20 款开源软件，并因此引入了 500 余个已知开源软件漏洞，其中包括超危漏洞 3 个、高危漏洞 8 个。例如，该 OA 系统中使用的 Spring Framework 3.2.0.RELEASE 中，存在超危漏洞 CVE-2018-1270，可导致远程代码执行。该 OA 系统中使用的部分开源软件及漏洞情况如下表所示。

序号	开源软件名称	开源软件版本	漏洞情况
1	Spring Framework	3.2.0.RELEASE	超危:1，高危:2，中危:14
2	Apache Log4j	1.2.14	超危:1，低危:1
3	Apache Commons FileUpload	1.2.2	超危:1，高危:3，低危:1
4	MySQL	5.0.20	高危:2，中危:14，低危:454
5	Apache Standard Taglib Implementation	1.1.2	高危:1
6	jQuery	1.8.3	中危:6，低危:1
7	Select2	3.5.1	中危:1
8	bootstrap-select	1.4.3	中危:1
9	Ueditor	1.3.5	中危:1
10	Apache Commons Codec	1.7	低危:1

OA 系统是每个现代企业必不可少的重要信息系统之一，其自身的安全性对于整个企业的信息安全至关重要，OA 系统中由于开源软件漏洞而导致的软件供应链安全风险应该被企业客户了解和关注。

2、国内某流行 Windows 桌面软件供应链安全分析

本实例中分析的是国内流行的某跨平台即时通讯软件，该软件具备发送语音短信、视频、图片和文字等功能，国内用户数量巨大。

经分析发现，该即时通讯软件的 Windows 桌面版本，使用了 Google V8 JavaScript Engine、zlib、OpenSSL、libjpeg、FFmpeg、SQLite、libpng 等在内的 80 余款开源软件，存在已知开源软件漏洞上百个，其中超危漏洞 10 个，高危漏洞 26 个。例如，zlib 的超危漏洞 CVE-2016-9841，Google V8 JavaScript Engine 的超危漏洞 CVE-2016-2843。

该软件中使用的部分开源软件及漏洞情况如下表所示。

序号	开源软件名称	开源软件版本	漏洞情况
1	Google V8 JavaScript Engine	3.29.88.17	超危:4，高危:8，中危:4
2	zlib	1.2.8	超危:2，低危:2
3	SQLite	3.10.2	超危:4，高危:6，低危:9
4	SQLite	3.26.0	超危:3，高危:3，低危:8
5	libpng	1.5.1	高危:3，中危:11，低危:5
6	OpenSSL	1.1.1d	高危:3，中危:3
7	curl	7.60.0	高危:2，中危:6，低危:14
8	OpenSSL	1.1.1c	高危:2，中危:5，低危:2
9	FFmpeg	3.2	高危:1，中危:6，低危:9
10	libpng	1.6.37	中危:1

即时通信软件已经成为我们日常生活中不可缺少的沟通工具，其软件供应链安全风险可能会影响到我们每个人，应该引起足够的重视。

3、某国产网络设备固件供应链安全分析

本实例中分析的是某国产无线路由器设备，该设备支持远程脱机下载、远程文件共享、打印机共享等功能，还支持通过 3G/4G USB 网卡上网，其固件可在官网公开下载。

经分析发现，该设备固件基于 Busybox 开发，使用了 Lighttpd、Samba、zlib、SQLite、OpenSSL、PCRE、json-c、curl 等 60 余款开源软件，存在已知开源软件漏洞 260 多个，其中超危漏洞 6 个，高危漏洞 42 个。例如，Samba 的超危漏洞 CVE-2020-1472，Wget 的超危漏洞 CVE-2017-13090。

该路由器中使用的部分开源软件及漏洞情况如下表所示。

序号	开源软件名称	开源软件版本	漏洞情况
1	Samba	3.5.8	超危:3，高危:2，中危:19，低危:8
2	Samba	3.0.33	超危:2，高危:1，中危:16，低危:8
3	PCRE	8.31	超危:1，高危:19，中危:5，低危:4
4	SQLite	3.72	超危:1，高危:4，中危:9，低危:30
5	Wget	1.16	超危:1，高危:4，低危:5
6	curl	7.21.7	高危:6，中危:60，低危:47
7	libexpat	2.0.1	高危:3，中危:11
8	busybox	1_17_4	高危:2，中危:11，低危:3
9	OpenVPN	2.3.2	高危:2，中危:7，低危:4
10	vsftpd	2.0.4	高危:1，中危:2

无线路由器已成为当前每个单位及家庭的必需品，一旦攻击者利

用已知开源软件漏洞入侵路由器，可能会造成单位、家庭及个人敏感信息泄露，网络被控制等严重后果。

4、Google Chrome 浏览器供应链攻击实例分析

本实例分析中，我们验证了通过利用 SQLite 漏洞，可以成功攻击 Google Chrome 浏览器。该实例中的软件供应链风险传播链条如下：SQLite 被 Chromium 所使用，而 Google Chrome 浏览器基于 Chromium 开发。因此 SQLite 的漏洞影响了 Google Chrome 浏览器，导致可针对 Google Chrome 浏览器成功实施软件供应链攻击。

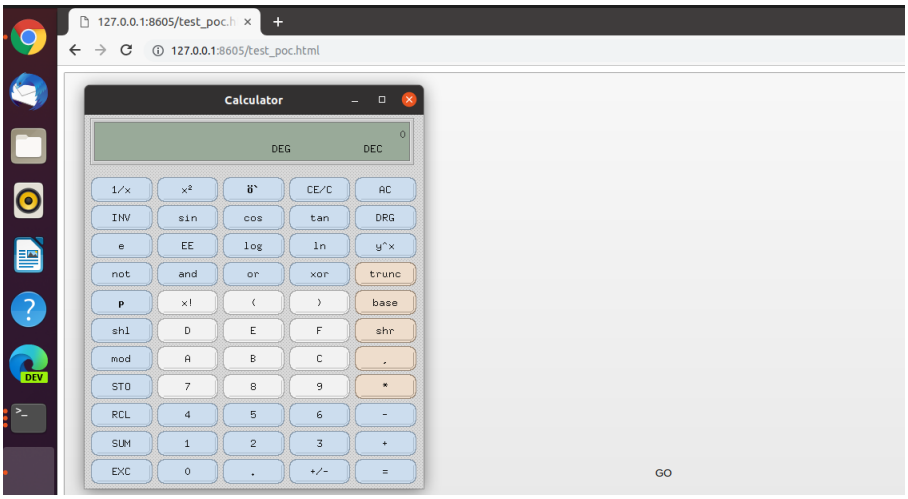
SQLite 是一款 C 语言编写的遵循 ACID 的开源关系数据库管理软件。由于 SQLite 实现了大多数 SQL 标准，且代码体积小，常常被用于到其他软件（如浏览器、操作系统、嵌入式系统）中。

Chromium 是 Google 为发展浏览器 Google Chrome 而发布的免费开源软件项目，一些其他浏览器也基于 Chromium，例如微软 Edge 浏览器、Opera 浏览器等。Chromium 中使用了大量的开源软件，包括 libpng、libxlst、zlib、Expat、SQLite 等超过 230 余款，其中 SQLite 被用于管理其 WEB SQL 数据库。

SQLite 在 2018 年到 2020 年间一共被发现 30 余个漏洞，以 CVE-2018-20346 为例，SQLite(版本<3.25.3)的 fts3 扩展中存在整数溢出漏洞，攻击者只需构造一系列 SQL 语句，即可实现内存的越界读写，造成远程代码执行的危害。

攻击者可以将漏洞利用代码隐藏在网页中，并诱导用户访问网页。

当用户使用 Google Chrome 浏览器（版本<=70.0.3538.77）访问包含漏洞利用代码的网页时，将会遭受到攻击，自动执行攻击者所设定的程序。下图为成功攻击的演示示例，当用户使用浏览器访问特定网页时，网页中隐藏的漏洞利用代码自动执行，调用和弹出计算器程序。

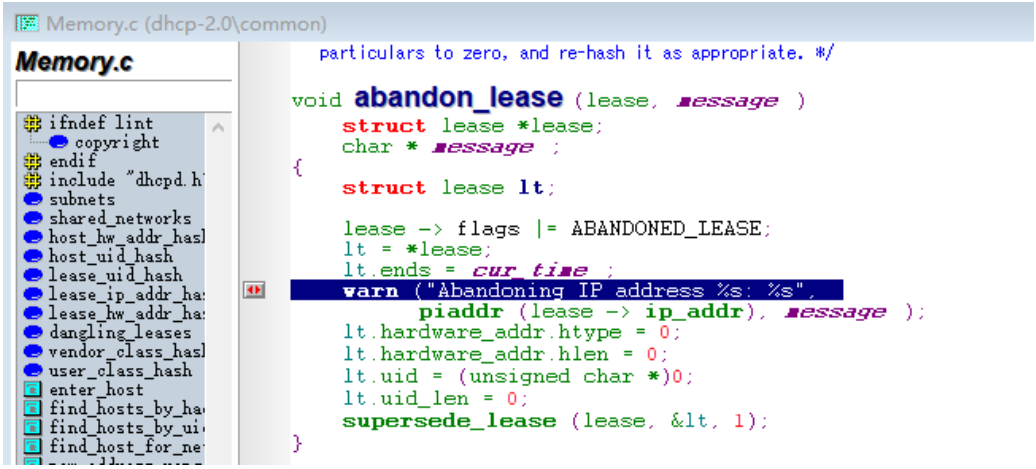


5、VMware Workstation 供应链攻击实例分析

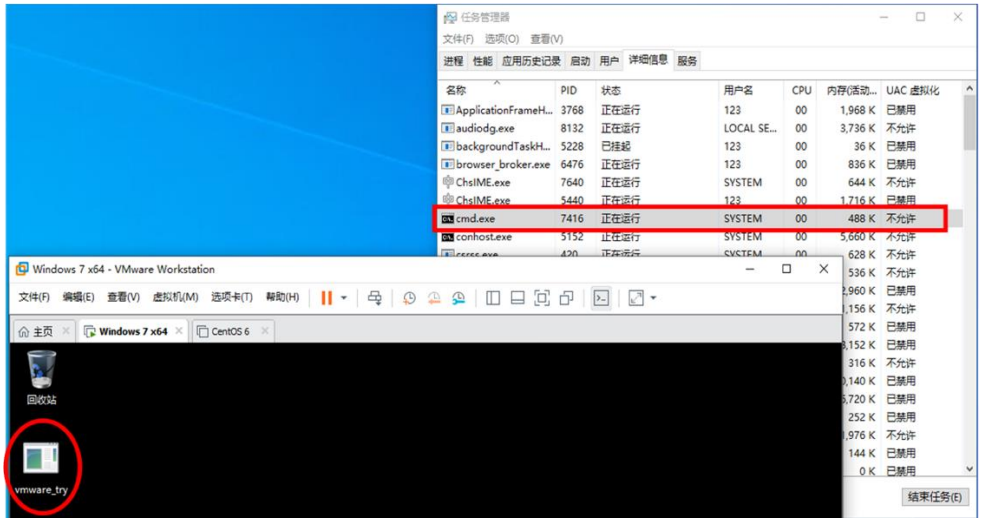
CVE-2020-3947 是一个存在于 VMware Workstation 和 VMware Fusion 产品中 vmnetdhcp 服务里的 UAF 漏洞，VMware 官方评估这是一个极其严重的漏洞，CVSSv3 评分为 9.3。

奇安信代码安全实验室分析发现，该漏洞实际是开源软件 ISC DHCP 2 代码中的漏洞，VMware 公司在其 vmnetdhcp 服务程序中使用了 ISC DHCP 2 的代码，从而导致其 VMware Workstation 和 VMware Fusion 产品中引入了该漏洞。此漏洞产生在 ISC DHCP 2 中 Memory.c 源代码文件的 abandon_lease 函数。此函数在第三行进行结构体赋值操作（“lt = *lease”），然而结构中包含了对象指针 uid，后面 supersede_lease 将 lease 的 uid 释放了，但是在 lt 中还存有包含

此对象 uid 的指针，并且 supersede_lease 中对 lt 中 uid 进行了操作，从而导致了 UAF。如下图所示。



此漏洞可被利用于虚拟机穿透，危害极其严重。攻击者可以通过在客户机发送精心构造的 DHCP 包来攻击宿主机上的 vmnetdhcp.exe 进程，从而在宿主机上执行任意代码，实现虚拟机穿透效果。下图为攻击成功的演示示例，该演示示例的目标是通过在客户机中执行攻击测试程序，实现在宿主机中自动调用执行 cmd.exe，从而实现虚拟机穿透。如下图所示，在 VMware Workstation 15.5.0 中的 Windows 7 虚拟机（客户机）中执行 vmware_try 攻击测试程序之后，宿主机的 Windows 10 中 cmd.exe 被自动执行，虚拟机被穿透。



六、总结及建议

软件供应链已经成为网络空间攻防对抗的焦点，直接影响关键基础设施和重要信息系统安全。为了应对软件供应链安全挑战，美国总统拜登在2021年5月12日签署了“加强国家网络安全的行政命令”，明确提出要增强美国联邦政府的软件供应链安全，要求向美国联邦政府出售软件的任何企业，不仅要提供软件本身，还必须提供软件物料清单（SBOM），明确该软件的组成成分。行政命令中还要求美国国家标准与技术研究院（NIST）在6个月内发布软件供应链安全指南，并在1年内发布最终指南。该行政命令被认为是迄今为止美国联邦政府为保护美国软件供应链安全采取的最强劲措施。

当前，我国在软件供应链安全方面的基础比较薄弱，亟需从国家、行业、机构、企业各个层面建立软件供应链安全风险的发现能力、分析能力、处置能力、防护能力，整体提升软件供应链安全管理的水平。现代软件的供应链非常复杂，软件供应链安全管理是一个系统工程，需要长期持续的建设。基于奇安信代码安全实验室的研究和实践，我们建议可从以下方面入手开展软件供应链安全相关工作，并在此基础之上不断增强和完善。

1、从国家与行业监管层面，建议：

(1) 制定软件供应链安全相关的政策要求、标准规范和实施指南，建立长效工作机制。

(2) 建立国家级/行业级软件供应链安全风险分析平台，具备系统

化、规模化的软件源代码缺陷和后门分析、软件漏洞分析、开源软件成分及风险分析等能力，为关键基础设施、重要信息系统用户提供日常的自查服务，及时发现和处置软件供应链安全风险。

(3) 在产品测评、系统测评等工作中纳入软件供应链安全的内容，针对软件的源代码、制成品、运行中的软件系统等进行软件供应链安全的测试和评估。

2、从软件最终用户层面，建议：

(1) 参照监管要求及业内优秀实践，明确本单位内部软件供应链安全管理的目标、工作流程、检查内容、责任部门，并赋予责任部门足够的权力。

(2) 在采购商业货架软件时，应充分评估供应商的安全能力，并与供应商签署安全责任协议，要求供应商提供其软件产品中所使用的第三方组件/开源组件的清单，并明确要求，一旦这些第三方组件/开源组件出现安全漏洞，供应商需同样承担安全责任，提供必要的技术支持。

(3) 在自行开发软件系统或委托第三方定制开发软件系统时，应遵循软件安全开发生命周期管理流程，针对软件源代码进行安全缺陷检测和修复，同时要重点管控开源软件的使用，建立开源软件资产台账，持续监测和消减所使用的开源软件的安全风险。

3、从软件厂商层面，建议：

(1)提高安全责任意识，将安全作为产品的基础属性来对待，严控产品的安全质量。

(2)建立清晰的软件供应链安全策略，明确本单位内部软件供应链安全管理的目标、工作流程、检查内容、责任部门，并赋予责任部门足够的权力。

(3)严格管控上游，尤其重点管控开源软件的使用，建立开源软件资产台账，建议采用可融入软件开发流程的开源安全治理工具，持续监测和消减所使用的开源软件的安全风险。

(4)严控自主开发的代码质量，建议采用可融入软件开发流程的软件源代码安全分析工具，持续检测和修复软件源代码中的安全缺陷和漏洞。

(5)建立完善的产品漏洞响应机制，包括产品漏洞信息的收集、漏洞报告渠道的建立和维护、漏洞补丁的开发和发布、客户侧漏洞应急响应和修复支持等。

附录：奇安信代码安全实验室简介

奇安信代码安全实验室是奇安信集团旗下，专注于软件源代码安全分析技术、二进制漏洞挖掘技术研究与开发的团队。实验室支撑国家级漏洞平台的技术工作，多次向国家信息安全漏洞库 （CNNVD）和国家信息安全漏洞共享平台 （CNVD）报送原创通用型漏洞信息并获得表彰；帮助微软、谷歌、苹果、Cisco、Juniper、Red Hat、Ubuntu、Oracle、Adobe、VMware、阿里云、飞塔、华为、施耐德、Mikrotik、Netgear、D-Link、Netis、ThinkPHP、以太坊、Facebook、亚马逊、IBM、SAP、NetFlix、Kubernetes、Apache 基金会、腾讯、滴滴等大型厂商和机构的商用产品或开源项目发现了数百个安全缺陷和漏洞，并获得公开致谢。目前，实验室拥有国家信息安全漏洞库（CNNVD）特聘专家一名，多名成员入选微软全球 TOP 安全研究者、Oracle 安全纵深防御计划贡献者等精英榜单。在 Pwn2Own 2017 世界黑客大赛上，实验室成员还曾获得 Master of Pwn 破解大师冠军称号。

基于奇安信代码安全实验室多年的技术积累，奇安信集团在国内率先推出了自主可控的软件代码安全分析系统——奇安信代码卫士和奇安信开源卫士。奇安信代码卫士是一套静态应用程序安全测试系统，可检测 1600 多种源代码安全缺陷，支持 C、C++、C#、Objective-C、Swift、Java、JavaScript、PHP、Python、Cobol、Go 等 20 多种编程语言。奇安信开源卫士是一套集开源软件识别与安全管控于一体的软件成分风险分析系统，通过智能化数据收集引擎在全球范围内广

泛获取开源软件信息和漏洞信息，帮助客户掌握开源软件资产状况，及时获取开源软件漏洞情报，降低由开源软件带来的安全风险，奇安信开源卫士目前可识别 4500 多万个开源软件版本，兼容 NVD、CNNVD、CNVD 等多个漏洞库。奇安信代码卫士和奇安信开源卫士目前已经在数百家大型企业和机构中应用，帮助客户构建自身的代码安全保障体系，消减软件代码安全隐患，并入选国家发改委数字化转型伙伴行动、工信部中小企业数字化赋能专项行动，为中小企业提供软件代码安全检测平台和服务。