公 开

2015 年第一季度开源软件源代码
安全漏洞分析报告

国家互联网应急中心

实验室

2015 年 4 月

目录

1 概述

近年来，随着软件技术飞速发展，开源软件已在全球范围内得到了广泛应用。数据显示，从 2012 年起，已有超过 80%的商业软件使用开源软件。开源软件的代码一旦存在安全问题，必将造成广泛、严重的影响。为了解当前开源软件的安全情况，实验室本季度对 31 款广泛使用的知名开源软件进行了源代码安全测试。结合漏洞扫描工具和人工审计的结果，形成了本漏洞分析报告。本次测试在代码层面共发现安全漏洞 52252 个，其中极易被攻击者利用的高危漏洞占比非常高，达到了总数的 40%，说明开源软件存在数量众多的安全问题。

2 被测开源软件

表 1 列出了本次被测的 31 个开源软件项目的概况，涵盖 C/C++，java，php 和 javascript 这四种编程语言。这些项目都是国际、国内知名的拥有广泛用户的软件项目的最新版本，而且其中不乏由知名软件公司开发的软件。由于这些软件大多具有巨大的用户群体，软件中的安全漏洞很可能会造成严重的后果。例如，Discuz!已拥有 14 年以上的应用历史和 200 多万网站用户案例，是全球成熟度最高、覆盖率最大的论坛软件系统之一。

表 1 被测开源软件项目概览

项目名称	版本号	主要编程语言	功能说明	代码行数(L)
Apache 日志分析	1.0	c/c++	分析 Apache 日志文件的工具	1345
ThinkAndroid	1.0	Java Android	Android 快速开发框架	31801
PwdManage	1.0	Java Android	Android 随身密码管理软件	4054
aFinal	1.0	Java Android	Android 快速开发框架	14106
YiIM	1.0	Java Android	Android 即时通讯	26237

android-explorer	1.0	Java Android	Android 文件管理器（雪梦）	6944
Argo	1.0	Java Web	58 同城的内部 Web 框架	22621
ja-sig cas	1.0	Java Web	单点登录系统	83326
师说 CMS	1.0	Java Web	一个内容管理系统	12734
Jfinal	1.9	Java Web	Java 极速 WEB+ORM 框架	20094
Dlog4j	1.0	Java Web	Java 多用户博客系统	62841
Shop++	3.0beta	Java Web	Java 开源网店系统	45909
Jeecms	6.0	Java Web	一个内容管理系统	7207
openfire	3.9.3	Java Web	即时通讯和群聊系统	386959
cynthia	1.0	Java Web	问题、BUG、任务、项目管理系统	72189
ECP	1.0	Java Web	中小企业客户关系进销存系统	30985
Druid	9.0	Java db	Jdbc 连接池、监控组件	287006
otter	4.2.0	Java db	阿里巴巴分布式数据库同步系统	99288
easywebsocket	1.0	Javascript	封装了 websocket API 的 javascript 库	34383
nodeBB	1.0	Javascript	Node.js 论坛系统	57056
jsGen	1.0	Javascript	Node.js 社区网站系统	63644
Webot	1.0	Javascript	微信公共账号机器人	491
cloudajs	1.0	Javascript	移动 WebApp 开发框架	29797
Codec2i	1.2.1	Php	国内首家开源众筹系统	18894
Discuz!	3.2	Php	开源论坛系统	261058
opencart	5.1	Php	开源电子商务系统	248585
wordpress	4.1	Php	著名博客平台	255890
Shop72hour	1.0	Php	开源微型导购系统	1672
zentaopms	6.4	Php	开源项目管理软件	106233
thinkphp	3.2.4	Php	轻量级 PHP 开发框架	74276
StartBBS	1.0	Php	轻量级开源社区系统	64516

3 测试内容

3.1 安全漏洞种类

本次测试涵盖各类常见安全漏洞。根据安全漏洞形成的原因、被利用的可能性、造成的危害程度和解决的难度等因素进行综合考虑，可以将常见的安全漏洞分为八类：

1) 输入验证与表示（Input Validation and Representation）

   输入验证与表示问题通常是由特殊字符、编码和数字表示所引起的，这类问题的发生是由于对输入的信任所造成的。这些问题包括：缓冲区溢出、跨站脚本、SQL 注入、命令注入等。

2) API 误用（API Abuse)

   API 是调用者与被调用者之间的一个约定，大多数的 API 误用是由于调用者没有理解约定的目的所造成的。当使用 API 不当时，也会引发安全问题。

3) 安全特性（Security Features）

   该类别主要包含认证、访问控制、机密性、密码使用和特权管理等方面的漏洞。

4) 时间和状态（Time and State）

   分布式计算与时间和状态有关。线程和进程之间的交互及执行任务的时间顺序往往由共享的状态决定，如信号量、变量、文件系统等。与分布式计算相关的漏洞包括竞态条件、阻塞误用等。

5) 错误和异常处理缺陷（Errors）

   这类漏洞与错误和异常处理有关，最常见的一种漏洞是没有恰当的处理错误（或者没有处理错误）从而导致程序运行意外终止，另一种漏洞是产生的错误给潜在的攻击者提供了过多信息。

6) 代码质量问题（Code Quality）

低劣的代码质量会导致不可预测的行为。对于攻击者而言，低劣的代码使他们可以以意想不到的方式威胁系统。常见的该类别漏洞包括死代码、空指针解引用、资源泄漏等。

7) 封装和隐藏缺陷（Encapsulation）

合理的封装意味着区分校验过和未经检验的数据，区分不同用户的数据，或区分用户能看到和不能看到的数据等。常见的漏洞包括隐藏域、信息泄漏、跨站请求伪造等。

8) 代码运行环境的缺陷（Environment）

该类漏洞是源代码之外的问题，例如运行环境配置问题、敏感信息管理问题等，它们对产品的安全仍然是至关重要的。

前七类漏洞与源代码中的安全缺陷相关，它们可以成为恶意攻击的目标，一旦被利用会造成信息泄露、权限提升等严重后果。最后一类漏洞描述实际代码之外的安全问题，它们容易造成软件的运行异常、数据丢失等严重问题。

3.2 安全漏洞级别

我们将源代码的安全问题分为四种级别：极高危（Critical）、高危（High）、中等（Medium）和低（Low）。衡量级别的标准包括两个维度，可信程度（confidence）和严重程度（severity）。可信程度是指发现的问题是否准确的可能性，比如将每个strcpy()调用都标记成缓冲区溢出漏洞的可信程度很低。严重程度是指假设测试技术真实可信的情况下检出问题的严重性，比如缓冲区溢出（buffer overflow）通常是比空

指针引用（null pointer dereference）更严重的安全问题。将这两个因素综合起来可以准确的为安全问题划分级别，如图 1 所示。

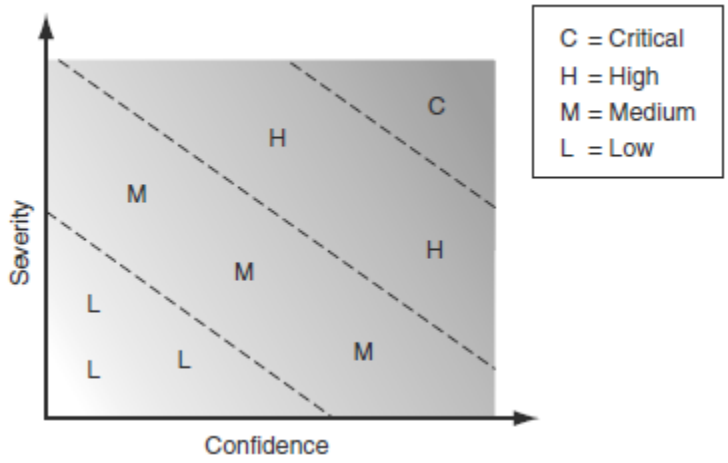


图 1 漏洞级别与严重程度、可信程度的关系

4 开源软件项目的安全漏洞情况

4.1 安全漏洞情况概览

由于极高危、高危级别的安全漏洞危害程度较高，更能反映出软件中迫切需要解决的安全问题，本部分仅讨论被测项目中的这两种级别的漏洞情况。表 2 展示了被测开源项目中存在的极高危以及高危安全漏洞的情况，图 2 用柱状图展示了该表中的各项分析结果。从中可以看出，大多数软件项目都存在不同程度的安全问题。这些项目中总计发现极高危漏洞 10544 个，高危漏洞 10438 个，有的项目（比如 Discuz、Openfire 和 Wordpress）的漏洞总数甚至超过了 2000 个。

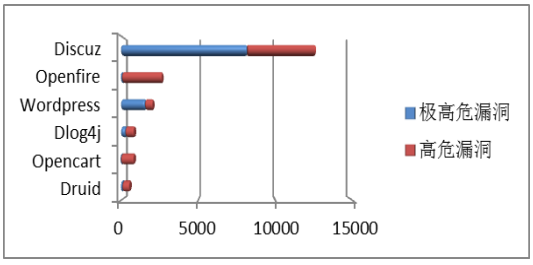
漏洞数量最多的软件项目是 Discuz，同时它也是漏洞密度（漏洞数量/代码行数）最高的软件。作为国内非常流行的内容管理型 PHP 程序，其规模庞大，开发人员众多，素质参差不齐，同时相对其他语言而言，PHP 本身也较容易产生安全缺陷，这些都有可

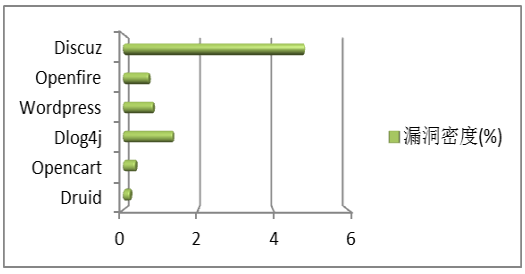
能是导致其安全问题突出的原因。漏洞数量排名第 2 的 Openfire 项目是 JAVA WEB 程序，该项目拥有高达 38 万行的代码量，是本次被测项目中代码量最大的软件。漏洞数量排名第 3 的 Wordpress 也是规模庞大的内容管理型 PHP 程序，但其漏洞密度却控制的相对较好，只有 0.8%，仅相当于同类型的 Discuz 的五分之一。这些统计结果展示了这批项目的安全性情况，排名靠前的项目处于极其容易被攻击者利用的状态，实际使用者急需通过安装补丁或者更新版本的方式进行修复和升级。本次测试的样本数据显示，随着软件规模的增大，安全漏洞数目会随之更快速的升高；PHP 程序比其他语言更容易充斥安全漏洞；同时，国外软件（如 wordpress）在安全缺陷控制上比国内软件（如 discuz）更加规范。

表 2. 开源软件项目中安全漏洞概览

项目名称	极高危(C)	高危(H)	总和(H+C)	漏洞密度%((H+C)/L)
Discuz	8167	4483	12650	4.845666
Openfire	65	2590	2655	0.686119
Wordpress	1556	501	2057	0.803861
Dlog4j	286	549	835	1.32875
Opencart	3	822	825	0.331878
Druid	93	449	542	0.188846
Codec2i	186	51	237	1.254366
Ja-sig cas	27	150	177	0.212419
Startbbs	58	62	120	0.186
Otter	9	92	101	0.101724
Jfinal	2	96	98	0.487708
zentaopms	25	66	91	0.085661
Cynthia	38	50	88	0.121902

Shop++	4	68	72	0.156832
Easywebsocket	0	59	59	0.171596
Ecp	2	57	59	0.190415
Jsgen	0	49	49	0.076991
Thinkadroid	0	45	45	0.141505
Jeecms	0	41	41	0.568891
Shop72hour	16	14	30	1.794258
YiIM	0	27	27	0.102908
Clouda	1	24	25	0.083901
Thinkphp	6	19	25	0.033658
Argo	0	21	21	0.092834
Nodebb	0	19	19	0.033301
Pwdmanage	0	14	14	0.345338
师说 cms	0	13	13	0.102089
android-explorer	0	8	8	0.115207
Afinal	0	6	6	0.042535
Webot	0	1	1	0.203666
Apache 日志分析	0	0	0	0





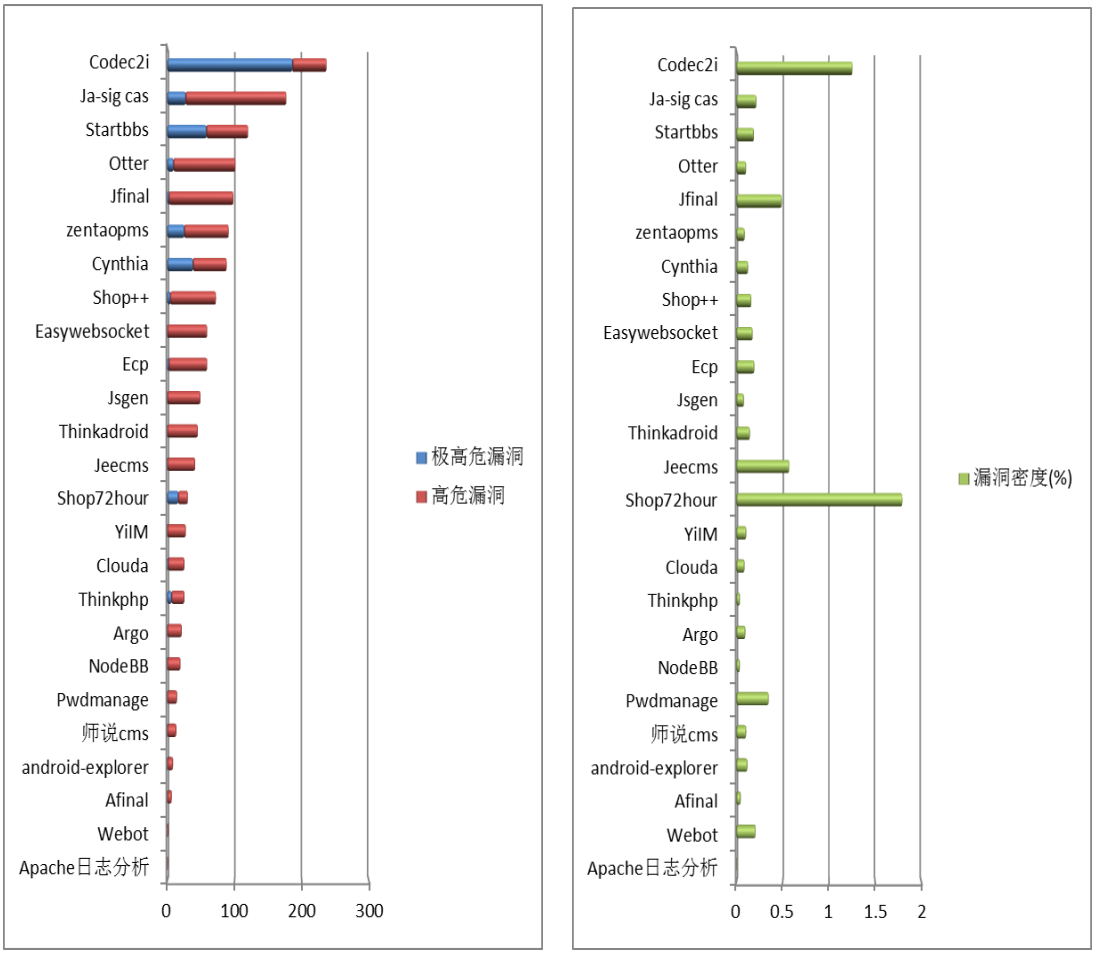


图 2 开源软件项目中安全漏洞分析图

4.2 安全漏洞总体分布情况

本部分进一步展示被测项目中，不同类型的安全漏洞的总体分布情况。图 3 和图 4 展示了被测项目中安全漏洞大类（根据 3.1 节划分）的分布情况，其中图 3 汇总了全部四种级别的安全漏洞，图 4 仅统计了高危以上级别的安全漏洞。对比图 3 和图 4 的数据可以发现，最常见的漏洞是“输入验证与表示”类漏洞，且该类漏洞的危害性较高，易被攻击者利用，通过绕过用户输入验证从而对 Web 应用系统进行攻击。此外，“错误和异常处理”、“封装和隐藏”两类漏洞在高危级别中都大幅降低，表明这两种类型的漏洞相对来说威胁较低，容易被开发人员忽视；这些漏洞虽然不易直接产生重大危害，

但可能导致系统运行不稳定、系统重要信息泄露等安全隐患，一旦被攻击者利用也会造成严重后果。

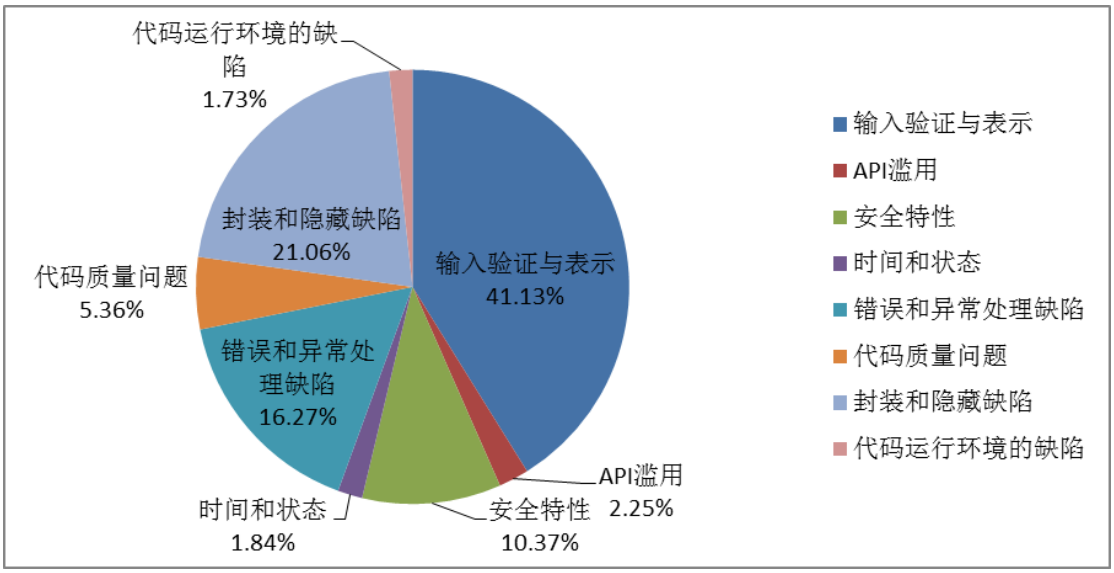


图 3 被测项目中的全部安全漏洞的分布情况（按大类划分）

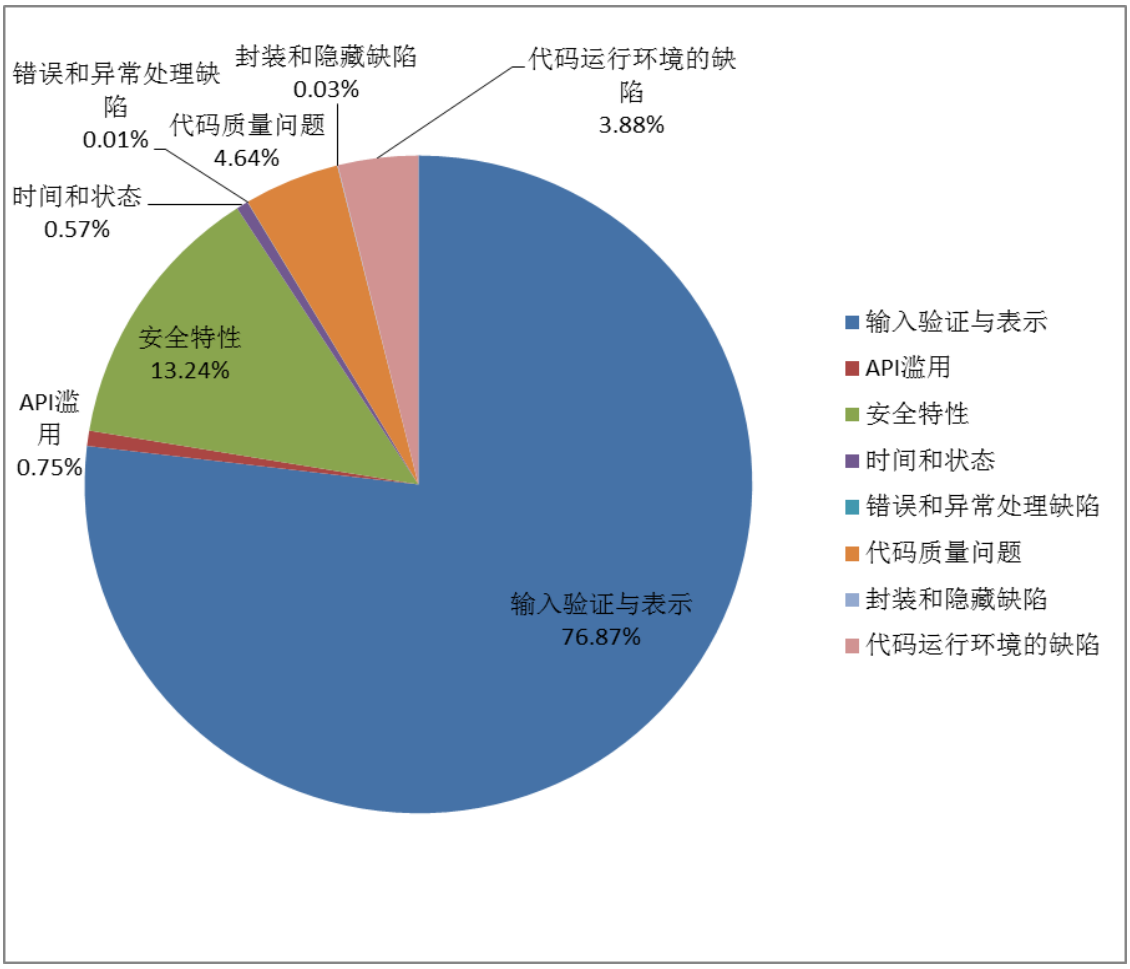


图 4 被测项目中的高危和极高危安全漏洞的分布情况（按大类划分）

图 5 和图 6 进一步展示了被测项目中的各种具体的安全漏洞的分布情况，其中图 5 汇总了全部四种级别的安全漏洞，图 6 仅统计了高危以上级别的安全漏洞。在被测的 31 个项目中，出现的比例超过 2%的漏洞共有 12 种。可以看出，在全部的漏洞中，出现较多且危害较严重的漏洞包括跨站脚本（共 8829 个，其中高危以上的 5783 个）、命令注入（共 3178 个，其中高危以上的 3123 个）、路径操纵（共 3083 个，全部是高危以上漏洞）。此外，错误处理不当（8500 个）、系统信息泄露（4536 个）两种漏洞虽然不易直接产生重大危害，但容易被开发人员忽视，出现数量较多；这些漏洞会影响系统稳定性，一旦被攻击者利用也会造成严重后果。下面对出现较多的几种漏洞进行简要说明，并给出防范建议。

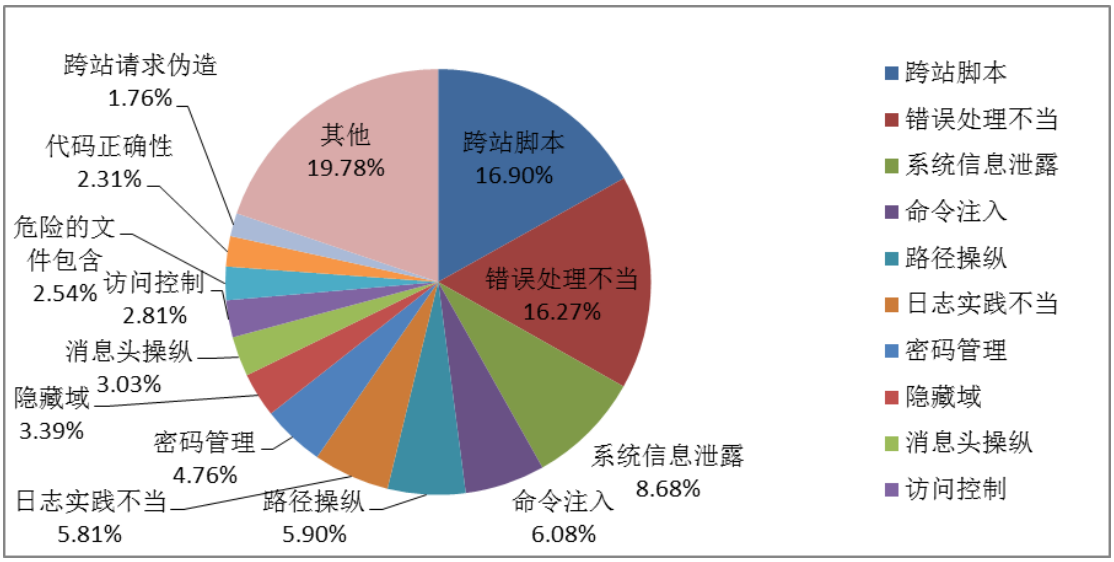


图 5 被测项目中的全部安全漏洞的分布情况（按具体安全漏洞划分）

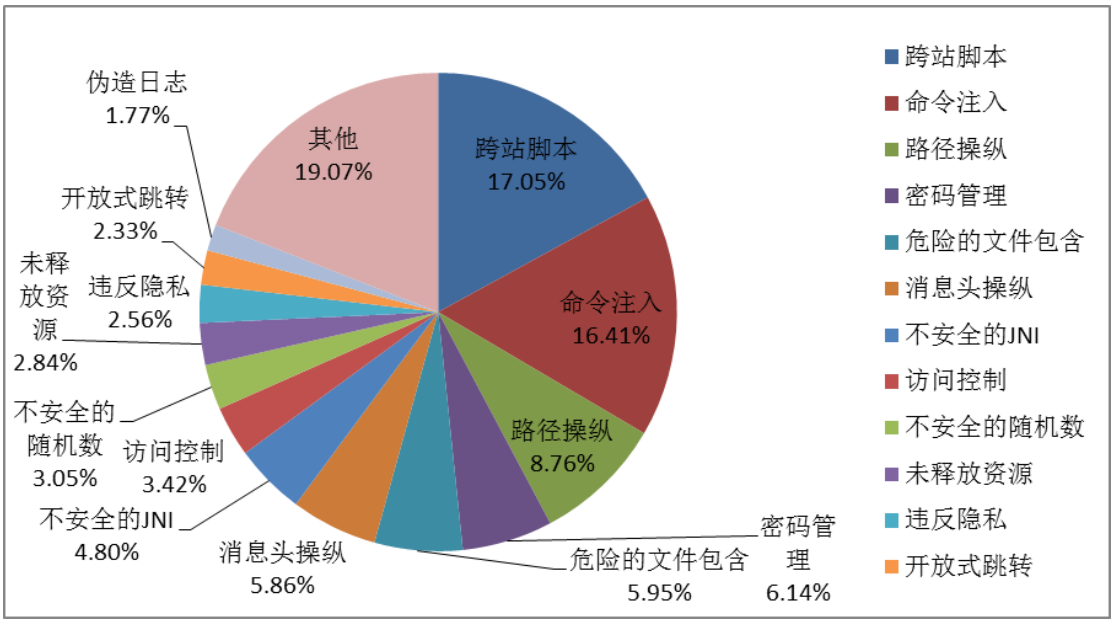


图 6 被测项目中的高危和极高危安全漏洞的分布情况（按具体安全漏洞划分）

1）跨站脚本（属于输入验证与表示类漏洞）

危害：向一个 Web 浏览器发送未经验证的数据会导致该浏览器执行恶意代码。

防范：验证所有输入数据，有效检测攻击；对所有输出数据进行适当的编码，以防止任何已成功注入的脚本在浏览器端运行。

2）错误处理不当（属于错误和异常处理缺陷）

危害：对于提供了错误或异常处理 API 的程序语言，如果对于错误不做处理或处理不当，会造成输出大量错误信息、程序崩溃等后果。

防范：遵守错误处理规范，错误处理要全面，尤其对于处理过程中出现的新错误也要进行嵌套式处理。

3）系统信息泄露（属于封装和隐藏缺陷）

危害：会暴露系统数据或调试信息，这会帮助攻击者获悉系统脆弱点，从而进行攻击的尝试。

防范：区分敏感和不敏感信息，对于敏感信息禁止输出。

4）命令注入（属于输入验证与表示类漏洞）

危害：攻击者将命令通过用户输入传递给程序，一旦绕过验证，则会执行命令，从而达到提权、获取敏感信息的目的。

防范：禁止程序执行用户可控的命令。

5）路径操纵（属于输入验证与表示类漏洞）

危害：允许用户输入访问文件系统的路径，可以使攻击者访问或修改受保护的系统资源。

防范：验证与路径访问相关的所有输入数据，采用恶意路径匹配等方式有效检测攻击。

6）密码管理（属于代码质量大类）

危害：程序员将密码嵌在代码逻辑当中，不仅带来逻辑上的难于理解和不好维护，也可能造成敏感信息泄露。

防范：密码应以加密方式保存在数据库或项目配置文件中。

7）危险的文件包含（属于输入验证与表示类漏洞）

危害：许多现代网络编写语言都能够在一个封装的文件内包含附加的源文件，从而使代码可以重用和模块化。被包含的文件都会作为主文件的一部分进行解析，并采用相同的方式来执行。当未验证的用户输入控制了所包含文件的路径时，可能导致恶意代码的执行。

防范：验证与文件包含相关的所有输入数据，采用恶意路径匹配等方式有效检测攻击。

5 关于本报告的说明

一、本报告仅从代码角度进行漏洞分析。在实际系统中，由于软件实际部署环境、安全设备等的限制，部分漏洞可能无法通过渗透测试得到验证。

二、本报告中的漏洞仅适用于表 1 中列出的特定软件版本。当软件版本有任何更新、修改和优化时，本报告不再适用。