系统安全防护中的业务逻辑漏洞检测与防御策略

肖帅帅[1]，蔡晶晶[2]，郭敏[1]，安达[1]

（1. 浙江东安检测技术有限公司，杭州 310013；2. 浙江广播电视集团，杭州 310005）

摘　要：近年国内多次开展系统安全防护，从攻击者的角度帮助防守方发现系统中存在的漏洞，其中多以服务器、中间件、第三方组件等通用漏洞和 SQL 注入、XSS 等常规漏洞为主。随着通用漏洞和常规漏洞被整改修复和系统得到安全设备的防护，此类漏洞在系统安全防护中明显减少；而业务逻辑漏洞具有攻击特征少和不可预见性的特点，很难通过传统安全设备发现，业务逻辑漏洞必将成为攻击者的主要突破口。文章针对业务逻辑漏洞的产生原因，总结了一套业务逻辑漏洞的检测流程，以电商平台的业务场景进行实例分析，演示业务逻辑漏洞的危害，最后，介绍了一些有针对性的防御策略。

关键词：系统安全防护；业务逻辑；漏洞检测；防御策略词

0 引言

近年国内通过以攻促防的手段，显著提高了国内信息系统的安全防护能力，也强化了技术人员的安全意识。系统安全防护通过模拟攻击者的方式来帮助防守方发现系统中存在的漏洞，防守方在实际对抗过程中既可以识别自身系统的风险点，又可以熟悉攻击者的思维和攻击手法。

借助自动化工具扫描发现通用漏洞以获取服务器权限和弱口令进入系统[1]，结合文件上传、SQL注入等常规漏洞获取服务器权限，这两种方式是目前演练中最直接有效的攻击手段，但忽视了应用系统可能存在的业务逻辑漏洞。业务逻辑漏洞一般不一定会使得攻击者获得服务器权限，可能在演练过程中得分较低，但会造成敏感信息泄露、权限绕过等高危风险，这些漏洞会对实际业务生产环境造成更加直接的影响[2]。

本文针对业务逻辑漏洞的产生原因，总结了一套业务逻辑漏洞的检测流程，以电商平台的业务场景进行实例分析，演示业务逻辑漏洞的危害，同时介绍一些有针对性的防御策略。

1 业务逻辑漏洞

应用系统都是通过逻辑实现的各种功能，即使是简单的应用系统，每个业务流程都会涉及复杂的逻辑操作。业务逻辑漏洞[3]是由应用系统在程序设计和开发过程中的逻辑缺陷造成的，攻击者可以利用正常业务功能的非常规操作实现恶意攻击行为。与SQL注入、XSS、命令执行、目录遍历等常规漏洞[4]相比，业务逻辑漏洞具有攻击特征少和不可预见性的特点，很难通过部署传统的安全设备进行检测和防御[5]。

业务逻辑漏洞的危害往往与应用系统的业务场景相关[6]，如电商平台可能存在批量刷单、抢红包优惠券、修改

收稿日期：2021-06-27

作者简介：肖帅帅（1993—），男，安徽，本科，主要研究方向为信息系统业务安全；蔡晶晶（1991—），女，浙江，硕士，主要研究方向为网络安全拟态技术；郭敏（1986—），男，浙江，本科，主要研究方向为网络安全检测、网络安全等级保护测评；安达（1992—），男，浙江，硕士，主要研究方向为密码协议及 TPM 应用。

通信作者：蔡晶晶 netbeanscai@163.com

订单金额等漏洞；社交平台可能存在会员充值、聊天信息泄露等漏洞；游戏平台可能存在账户被盗、游戏作弊等漏洞。严重的业务逻辑漏洞还会威胁到应用系统的正常运行，造成大规模用户信息泄露。业务逻辑漏洞攻击一般较为隐蔽且难以被察觉，往往系统被攻击后很久才被发现，为系统的安全防护带来了极大挑战。

分析业务逻辑漏洞要结合具体的业务场景，不同的业务场景包含不同的功能模块，其业务流程也丰富多样，很难列举所有的业务逻辑漏洞。

业务逻辑漏洞可以大致分为3类。

1）权限控制类：未授权访问、水平／垂直越权、账号关联覆盖等；

2）功能缺陷类：无效的反自动化手段、缺少数据并发读写锁（竞争写入漏洞）、请求重放、失效的签名算法等；

3）校验不当类：变量覆盖、上下限校验、校验标准不一致、缺少绑定校验、业务流程乱序等。

2 业务逻辑漏洞检测

系统安全防护主要是在信息系统外部模拟攻击者发现系统漏洞，一般采用黑盒测试，由于缺少系统内部权限和技术资料，需要测试人员主动收集应用系统信息，了解被测系统功能模块、业务流程、数据处理、安全规则等。业务逻辑漏洞检测[7]可分为4个阶段：业务场景建模、业务流程梳理、风险点识别和检测分析。

2.1 业务场景建模

通过手工点击熟悉应用系统的各个功能，了解功能模块之间的相互联系。需特别关注不同行业、平台的应用系统有其专属业务场景的功能特性，如电商平台的订单管理、视频网站的会员充值等功能。

以电商平台的业务场景建模为例，如图1所示。

2.2 业务流程梳理

应用系统场景建模后，需要对系统各个功能模块的业务流程进行梳理，识别功能实现的业务逻辑、数据包响应处理过程、关键参数字段的作用等。用户在电商平台选购商品下单并完成支付的整个交易过程如图2所示。

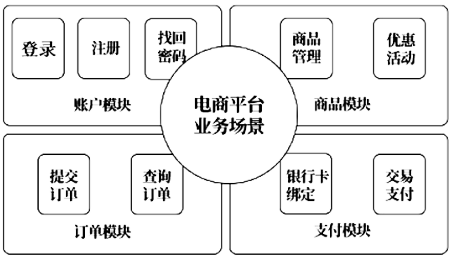

图 1 电商平台业务场景模型

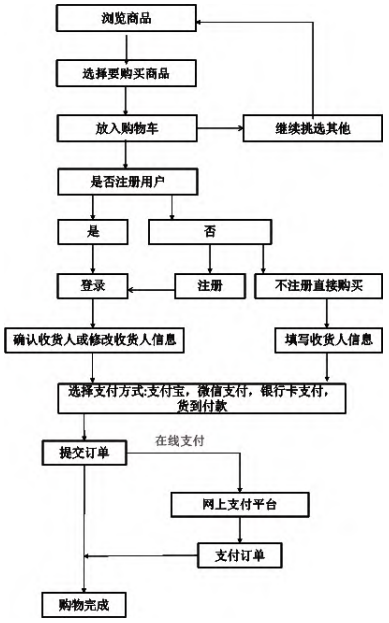

图 2 商品订单支付流程

2.3 风险点识别

在完成业务场景建模和业务流程梳理后，分析应用系统各功能模块每一步操作可能存在的业务风险，尽可能详细地列举各业务流程中的风险点，有助于下一阶段综合全面的检测分析。

以电商平台的主要功能模块为例，风险点识别结果如表1所示。

2.4 检测分析

检测分析[8]前还应尽可能多地收集应用系统资料信息，为检测提供足够的分析依据，可通过网络爬虫发现应用系统的路径、静态页面、可执行文件、表单、隐藏接口、JS文件等；收集整理应用系统资产、服务器操作系统、中间件、数据库、第三方组件等信息；查看公开的应用系统介绍文档、技术资料、操作指南

表 1 电商平台业务风险点

功能模块	子功能	业务风险点
账户模块	登录	用户名密码暴力破解 账户信息泄露 / 撞库 验证码爆破和绕过
	注册	恶意用户批量注册 恶意验证注册账户 已有用户名覆盖
	找回密码	重置任意用户密码 短信验证码劫持 用户邮箱劫持篡改
商品模块	商品管理	刷单好评 异常定价 违规上货
	优惠活动	盗刷积分 抽奖作弊 批量获取优惠券 / 代金券 更改优惠券 / 代金券金额、数量
订单模块	提交订单	更改订单价格 突破限购规则
	查询订单	订单信息泄露 账户信息泄露
支付模块	银行卡绑定	盗刷他人银行卡 银行卡信息泄露
	交易支付	多笔订单一次支付 订单支付绕过

等，任何围绕系统相关的信息都可能为业务逻辑漏洞检测提供帮助。

检测分析步骤如下：

1）通过代理抓包观察应用系统操作过程的数据请求包和响应包，分析存在哪些参数[9]；

2）依次测试每一个参数，从请求包中删除参数的名称和值，分析系统的异常响应或错误提示；

3）依次改变请求包中的参数值，使用递增值、空值、负值、极值、编码值等，观察参数值变化对应用系统的影响；

4）模拟不同用户对同一组数据进行操作，提取某个用户提交的所有参数，再由另一名用户提交这些参数，如果应用系统接收并处理这些参数，即可能存在权限控制类漏洞[10]；

5）尝试访问未知路径、输入特殊字符、改变请求方式等操作引发应用系统错误，分析系统的错误处理逻辑；

6）竞争条件测试[11]，对同一功能进行多线程大批量的业务操作请求，分析结果是否存在异常；

7）分析参数值的变化规律，猜测应用系统使用的算法，模拟生成参数值如cookie、session、token、验证码等，替换请求包的参数值，测试是否依然有效；

8）调整系统功能的业务流程执行顺序[12]，延时、跳过或重复某个操作步骤，检验系统是否存在业务流程乱序漏洞。如网上购物需要3步业务流程，如下单-支付-发货，尝试能否绕过支付环节直接进入发货环节。

3 业务逻辑漏洞实例

前面已经完成了对电商平台的业务场景建模、业务流程梳理和风险点识别，这里选择两个功能模块[10]的风险点来演示如何进行业务逻辑漏洞的检测分析。

3.1 重置任意用户密码业务逻辑漏洞检测分析

1）通过代理的方式登录电商平台，点击“忘记密码”，输入用户名wiener，查看密码重置邮件，点击密码重置链接，完成密码重置。

2）查看密码重置过程的数据包，提交新密码的POST /forgot-password?temp-forgot-password-toke请求包如图3所示，可以看到，请求体中有username参数。

```
POST /forgot-password?temp-forgot-password-token=jgPXtpaiB76yYzHQSqg9KizGax8YspT6
 HTTP/1.1
Host: ac4e1fb41e83694c803b026900f400d9.web-security-academy.net
Connection: close
Content-Length: 119
Cache-Control: max-age=0
sec-ch-ua: " Not;A Brand";v="99", "Google Chrome";v="91", "Chromium";v="91"
sec-ch-ua-mobile: ?0
Upgrade-Insecure-Requests: 1
Origin: https://ac4e1fb41e83694c803b026900f400d9.web-security-academy.net
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/91.0.4472.77 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image
/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer:
https://ac4e1fb41e83694c803b026900f400d9.web-security-academy.net/forgot-password
?temp-forgot-password-token=jgPXtpaiB76yYzHQSqg9KizGax8YspT6
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en-US;q=0.8,en;q=0.7
Cookie: session=72nXgdZbCOXG2QTPsdciTTIXSQrS2Sa3

temp-forgot-password-token=jgPXtpaiB76yYzHQSqg9KizGax8YspT6&username=wiener&
new-password-1=123456&new-password-2=123456
```

图 3 提交 wiener 新密码请求包

3）尝试删除URL和请求体中的temp-forgot-password-token参数的值，密码重置功能仍然有效，说明提交新密码时，应用系统后端未检查令牌。

4）返回登录页面，重新点击“忘记密码”，输入用户名wiener，查看密码重置邮件，点击密码重置链接。代理抓包修改提交新密码的请求包，删除URL和请求正文中的temp-forgot-password-token参数的值，将

username参数改为carlos，然后发送请求。

5）登录页面使用用户名：carlos，密码：123456，成功登录应用系统。

3.2 更改订单价格业务逻辑漏洞检测分析

1）通过代理的方式登录电商平台，添加皮夹克到购物车，提交订单显示余额不足。

2）查看订单提交过程的数据包，分析POST /cart请求包中的关键参数，依次测试参数对应用系统的作用，确认productId是商品编号，redir是重定向页面，quantity是添加商品数量；

3）经多次测试发现，quantity的最大有效值为99。

4）尝试自动化持续提交99件商品到购物车，刷新查看总价，注意到当数额不断增加的过程中数值突然变为一个位数较大的负整数，并开始逐渐变化向0值靠近。分析此状况是由于数值超过编程语言允许的整数最大值（2,147,483,647）造成的。

5）掌握了应用系统的业务逻辑，接下来考虑漏洞的利用：只要控制添加的商品数量，使购物车的商品总价低于账户余额即可提交订单完成支付。

4 业务漏洞防范策略

由于业务逻辑漏洞与应用系统的业务场景关联性强，且具有攻击特征少和不可预见性的特点，没有通用的漏洞解决方案，很难通过部署传统的安全设备进行检测和防御。防御业务逻辑漏洞需要结合业务本身从代码及配置层面做好风险管理，开发和测试人员需熟悉各类业务流程中的关键风险点，制定有针对性的防御策略。

1）依据最小权限原则进行用户管理，完善身份鉴别机制，执行操作前必须验证用户是否具有相应的权限；

2）采用密码技术确保系统重要数据的完整性和保密性，防止关键参数和敏感数据被篡改、泄露；

3）对用户输入数据进行有效性校验，重要参数在后端服务器校验（如验证码、上传文件类型等）；

4）特定功能设置合理的参数值限制，如用户登录次数、订单等待时长、验证码验证次数、同时段单个用户请求数等；

5）严格限制业务流程，检查用户的每一步操作，确保业务按照正确的流程顺序执行，防止攻击者绕过、跳过或重复任何流程；

6）审核应用系统公开信息，避免敏感信息（如开发文档、API接口、程序错误提示等）的泄露。

5 结束语

本文以电商平台的业务逻辑漏洞为实例进行检测分析，阐述了业务逻辑漏洞对应用系统的危害。主流开发框架、承载的系统服务的漏洞通常可以借助最新版本扫描器发现，但是业务逻辑漏洞的源头均为业务开发设计中的逻辑缺陷和验证流程缺失，现阶段人工智能技术尚未普及，无法通过工具自动适配发现不同业务场景的逻辑漏洞，只能借助系统安全防护人工发现。通过定期系统安全防护对应用系统进行安全风险评估和安全测试，查找应用系统中的业务逻辑漏洞，有助于强化业务逻辑漏洞的防御能力，提高应用系统的整体安全防护能力。

参考文献：

[1] 倪一涛，陈咏佳，林柏钢 . 基于自动解混淆的恶意网页检测方法 [J]. 信息网络安全，2019，19（4）：37-46.

[2] 吴翰清 . 白帽子讲 Web 安全 [M]. 北京：电子工业出版社，2014.

[3] STUTTARD D，PINTO M. 黑客攻防技术宝典 Web 实战篇 [M]. 石华耀，傅志红，译 . 2 版 . 北京：人民邮电出版社，2018.

[4] 喻志彬，马程，李思其，等 . 基于 Web 应用层的 DDoS 攻击模型研究 [J]. 信息网络安全，2019，19（5）：84-90.

[5] 黄长慧，胡光俊，李海威 . 基于 URL 智能白名单的 Web 应用未知威胁阻断技术研究 [J]. 信息网络安全，2021，21（3）：1-6.

[6] 陈晓光，胡兵，张作峰 . Web 攻防之业务安全实战指南 [M]. 北京：电子工业出版社，2018.

[7] 薛楠凤 . 基于渗透测试的逻辑漏洞检测技术研究 [D]. 成都：电子科技大学，2018.

[8] 冯丹 . web 应用业务逻辑漏洞检测技术研究 [D]. 杭州：浙江工商大学，2017.

[9] AFFIA A O. A Black-box Methodology for Attacking Business Logic Vulnerabilities in Web Applications[D]. Tallinn: Tallinn University of Technology, 2021-05-29.

[10] SOMMER Michael. Business Logic Vulnerabilities[EB/OL]. https://portswigger.net/web-security/logic-flaws, 2021-05-29.

[11] SAAD E, MITCHELL R, MEUCCI M. OWASP Business Logic Vulnerability[EB/OL]. https://owasp.org/www-project-web-security-testing-guide, 2020-12-03.

[12] TAS G. Ten Business Logic Attack Vectors Business Logic[EB/OL]. https://information.rapid7.com/top-10-business-logic-vectors-whitepaper.html, 2021-05-29.