阅读管理与规划系统测试文档

测试报告

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2016.9.4 | 1.0 | 测试报告 | 曾鹤鸣 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 4

2. 测试结果摘要 4

3. 基于需求的测试覆盖 4

4. 基于代码的测试覆盖 4

5. 建议措施 4

6. 图 4

测试报告

# 简介

本测试用例用于测试REST API以及web应用和手机APP产生的结果是否符合预期并使代码中的漏洞尽可能地暴露，以达到修复漏洞的目的。

## 目的

测试代码漏洞并及时修复。

## 范围

此用例测试的范围包括Server端的REST API是否能正常运行，web应用以及iOS APP是否运作正常。

## 定义、首字母缩写词和缩略语

### URL

测试用例分类中的URL指访问api的url，且url皆省略了http以及ip、端口等内容，特此说明。

### 表格中为空

表格中为空表示该项为“无”。

### Header

表格中*前置条件*中的Header指http请求的header中包含key-value对：(X-Auth-Token, “xxxxxxxxxx”)

# 测试结果摘要

## Login

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 前置条件 | Query 参数 | 说明 | 测试输入(request body) | 预期结果 | 结果 |
| 无 | 无 | 正确的email，password组合 | {  "email": "xxx",  "password": "xxx"  } | 200 | 200 |
| 无 | 无 | 错误的email，password组合 | {  "email": "xxx",  "password": "xxx"  } | 401 | 401 |
| 无 | 无 | 访问无效链接，如 /auth/token（应为 /auth/tokens） |  | 404 | 404 |
| 无 | 无 | 错误的json body | {“emai”:”xxx”,”password”:”xxxx”} | 400 | 400 |
| 无 | 无 | 错误的json body | {“password”:”xxx”} | 400 | 400 |
| 无 | 无 | 错误的json body | {“email”:”xxx”} | 400 | 400 |
| 无 | 无 | 错误的json body | {“email”:”xxx”,”password”:”xxx”,} | 400 | 400 |
| 无 | 无 | 非法输入 | {“email”:”a or 1”,”password”:”xxx”} | 400 | 400 |
| 无 | 无 |  | {“email”:”xxx”,”password”:”a||1;”} | 400 | 400 |

## Logout

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 前置条件 | Query 参数 | 说明 | 测试输入(request body) | 预期结果 | 结果 |
| Header | 无 | 正确的header且url访问正确 | 无 | 204 | 204 |
| 无 | 无 | Logout时不带header | 无 | 401 | 401 |
| Header | 无 | Logout时带错误的header | 无 | 401 | 401 |
| Header | 无 | Logout时带别人的header | 无 | 204 | 204 |

## Search book

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 前置条件 | Query 参数 | 说明 | 测试输入(request body) | 预期结果 | 结果 |
| Header | ?title=Computer | Search by 书名或部分书名 | 无 | 200 | 200 |
| Header | ?author=Donald | Search by 作者名或部分作者名 | 无 | 200 | 200 |
| Header | ?isbn=8274665382 | Search by isbn10 | 无 | 200 | 200 |
| Header | ?isbn=82746653821 | Search by isbn10，但不满足isbn10的格式，比如有11位 | 无 | 结果为空 | Null |
| Header | ?isbn=8274665382091 | Search by isbn13 | 无 | 200 | 200 |
| Header | ?isbn=827466538209 | Search by isbn13，但不满足isbn13的格式，比如有12位 | 无 | 结果为空 | Null |

## Book shelf 书架

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 前置条件 | Query 参数 | 说明 | 测试输入(request body) | 预期结果 | 结果 |
| Header | 无 | List该用户书架中的书籍 | 无 | 200 | 200 |
| 无header | 无 | List该用户书架中的书籍，但未加header | 无 | 401 | 401 |
| header中使用过期的token | 无 | List该用户书架中的书籍，但header中使用过期的token | 无 | 401 | 401 |
| Header | 无 | 向书架中添加书籍 | Book\_id合法 | 200 | 200 |
| Header | 无 | 向书架中添加书籍 | Book\_id不存在 | 404 | 404 |
| Header | 无 | 向书架中添加书籍 | Book\_id已在该用户的书架中存在 | 409 | 409 |
| Header | 无 | 删除书架中的一本书籍 | Book\_id合法 | 200 | 200 |
| Header | 无 | 删除书架中的一本书籍 | Book\_id不存在 | 404 | 404 |
| Header | 无 | 删除书架中的一本书籍 | Book\_id并不在该用户的书架中 | 404 | 404 |
| Header | 无 | 删除书架中的一本书籍 | Book\_id已在该用户的书架中存在 | 409 | 409 |

## reading\_progress阅读进度

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 前置条件 | Query 参数 | 说明 | 测试输入(request body) | 预期结果 | 结果 |
| Header | 无 | List该用户当前的阅读进度 |  | 200 | 200 |
| 无header | 无 | List该用户当前的阅读进度 |  | 401 | 401 |
| header中使用过期的token | 无 | List该用户当前的阅读进度 |  | 401 | 401 |
| Header | 无 | 添加某本书当前的阅读进度 | Current\_page 大于上一次读到的页数，并且小于total page | 200 | 200 |
| Header | 无 | 添加某本书当前的阅读进度 | Current\_page 不是数字 | 400 | 400 |
| Header | 无 | 添加某本书当前的阅读进度 | Current\_page too large，大于总页数 | 400 | 400 |
| Header | 无 | 添加某本书当前的阅读进度 | Current\_page小于上一次读到的页数 | 400 | 400 |
| Header | 无 | 添加某本书当前的阅读进度 | 这本书不在我的书架中 | 404 | 404 |

## track读书计划

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 前置条件 | Query 参数 | 说明 | 测试输入(request body) | 预期结果 | 结果 |
| header | ?category\_ref=1 | 搜索track |  | 200，并且返回的tracks按fork数和star数降序排列 | 200 |
| header | ?title=my | 搜索track |  | 200，并且返回的tracks按fork数和star数降序排列 | 200 |
| header | ?category\_ref=3&start=0&count=5 | 搜索track |  | 200，返回该类别的五个track，并且返回的tracks按fork数和star数降序排列 | 200 |
| header | ?start=aaaa | query参数的值不是整数 |  | 400 | 400 |
| header | ?option=0 | list我的track |  | 200，返回我的所有track | 200 |
| header | 无query参数（option缺省） | list我的track |  | 200，返回我的所有track | 200 |
| header | ?option=1 | list我的track |  | 200，返回我的track中所有由我原创的track | 200 |
| header | ?option=2 | list我的track |  | 200，返回我的track中所有我fork来的track | 200 |
| header | 无 | 创建track | 合法输入 | 200 | 200 |
| header | 无 | 创建track | request body中包含除了title, image, category and stage之外的不可编辑的部分 | 400 | 400 |
| header | 无 | fork别人的track到自己 | track\_id合法 | 200 | 200 |
| header | 无 | fork别人的track到自己 | track\_id不存在 | 404 | 404 |
| header | 无 | fork别人的track到自己 | 自己的track | 403 | 403 |
| header | 无 | start别人的track | track\_id合法 | track\_id合法 | track\_id合法 |
| header | 无 | start别人的track | track\_id不存在 | track\_id不存在 | track\_id不存在 |
| header | 无 | start别人的track | 自己的track | 自己的track | 自己的track |
| header | 无 | 向track的原作者发送pull request的改进请求 | track\_id合法 | track\_id合法 | track\_id合法 |
| header | 无 | 向track的原作者发送pull request的改进请求 | track\_id不存在 | track\_id不存在 | track\_id不存在 |
| header | 无 | 向track的原作者发送pull request的改进请求 | 自己的track | 自己的track | 自己的track |

# 基于需求的测试覆盖

|  |  |
| --- | --- |
| 需求 | 测试结果 |
| 用户注册、登录 | 通过 |
| 管理员登录 | 通过 |
| 添加、移除书架中的书籍 | 通过 |
| 设置阅读进度 | 通过 |
| 管理员管理用户、书籍 | 通过 |
| 创建阅读计划 | 通过 |
| 加入、编辑阅读计划，对已有读书计划提出更改建议 | 通过 |
| 解锁和获得称号 | 通过 |
| 撰写书评 | 通过 |
| 对于书评进行评论或点赞 | 通过 |
| 在讨论区提出、回答问题 | 通过 |
| 管理员可以通过爬虫更新书籍数据 | 通过 |

# 基于代码的测试覆盖

|  |  |
| --- | --- |
| 代码 | 测试结果 |
| AdminAuthorsResource | 通过 |
| AdminBookResource | 通过 |
| AdminTranslatorResource | 通过 |
| AdminUserResource | 通过 |
| Application | 通过 |
| AuthResource | 通过 |
| BookResource | 通过 |
| BookReviewLongResource | 通过 |
| BookReviewShortResource | 通过 |
| BookshelfResource | 通过 |
| CommentLongResource | 通过 |
| CommentShortResource | 通过 |
| CommentTrackResource | 通过 |
| HonorResource | 通过 |
| MessageResource | 通过 |
| ReadingProgressResource | 通过 |
| TrackResource | 通过 |
| TrackReviewResource | 通过 |

# 建议措施

在项目开始的时候，应该制定相应的标准，编码标准，需求变更标准等，开发和测试人员严格按照标准进行，可以在后期减少因为开发，测试不一致而导致的问题，同时可以降低沟通成本。

发布版本的时候，正确布置测试环境，减少因为测试环境，测试数据库数据的问题而出现的无效bug。

开发人员解决bug的时候，填写bug原因以及解决方式，方便bug的跟踪。 开发人员在开发版本上发现bug，可以通知测试人员，因为开发人员发现的bug很有可能在测试版本上出现，而测试人员和开发人员的思路不同，有可能测试人员没有发现该bug，而且，这样可以保证发现的bug都能够被跟踪。

做好版本的控制，从开发版本，测试版本做好每个环节的版本控制。

# 图

测试中主要的漏洞出现在DAO层与Service层，主要是用Hibernate、Morphia、Jedis操作数据库时；由于对于其内部机制的不了解，导致调用其中方法时产生各类问题；而Service层由于逻辑错误等问题容易产生漏洞。