Day5 混合压测报告 —— 王嘉萍（测试岗）

日期：2026-11-17（周二）｜主题：安全加固与并发正确性
被测版本：master `4e48df4`（含李佳 Day5 实名三档 + Day6 AES 加密改造），本地 8080
环境：Java 21 / Spring Boot 3 / MySQL 8（bugedui），压测线程池 + Barrier 同步释放

一、结论总览

场景	内容	结果
M1	并发报名 10 + 并发退出 5 + 取消活动 1 同刻释放	**未通过**：取消接口 5000，本轮死锁 5 条
M2	同一用户 20 连点重复报名	**通过**：1 成功 + 19 次 4011，库中恰 1 行
M3	15 人并发评价同一目标（信用流水核对）	**未通过**：丢失更新，+30 只到账 +4
全量回归	77 条（含 Day5 新增安全 10 条 + 守门 2 条）	73 通过 / 0 失败 / 1 skip / 2 xfail / 1 xpass（24.5s）

当日新发现缺陷 3 个（BUG-6 / BUG-7 / BUG-8，见第四节），BUG-4、BUG-5 验证结案。

二、M1 混合并发（报名 + 退出 + 取消）

设计：活动 20 人上限（19 坑），预热 9 人占坑；同一 Barrier 释放 16 线程 ——
10 人报名（恰好填满余坑）、5 人退出（持预热时留存的 registrationId）、发起人取消活动。

指标	实测
返回码分布	报名 10 条全 0；退出 5 条全 0；**取消 1 条 5000**
本轮死锁（日志增量）	**5 条**，`DeadlockLoserDataAccessException`
死锁位置	`ActivityServiceImpl.cancelActivity` 第 383 行 `quitByActivityCancel` 批量 UPDATE
名额账本	current_people=15 = 14 条有效报名 + 发起人，**无超卖、无重复行**
延迟	avg 46~63ms，p95 69~77ms，总耗时 0.08~0.28s

复跑 4 次的形态：死锁的受害者会轮换（第一轮取消失败、另一轮 2 条退出失败 5000、第三轮取消又失败）——**锁序冲突随机打在其中一个事务上，MySQL 挑一个回滚**，所以表象不稳定，但死锁本身 4/4 全部复现。

根因：signup 事务"插 registration + UPDATE activity.current_people"，cancel 事务"UPDATE activity → 批量 UPDATE registration"，两个事务以相反顺序获取 `activity` 与 `registration` 行锁 → 环形等待。

BUG-6（P1）：混合并发下取消活动死锁回滚，接口返回 5000（应为 0）。
修复建议：统一加锁顺序（都先锁 activity 再动 registration），或缩小取消事务、
退名额改由取消提交后异步补偿。守门用例 `tests/test_08_concurrency.py::test_cc01`（xfail 挂账，
复现概率约 2/3，判定以 M1 压测为准）。

三、M2 同一用户并发重复报名（幂等验证）

指标	实测
20 连点返回	1 条 code=0 + 19 条 4011（请勿重复报名），0 条其它
库侧	registration 有效行恰 1 条，current_people=2（发起人+本人）
结论	**通过**：应用层防重快查 + `uk_*` 唯一索引兜底，无重复行、无超卖

幂等这一环是可靠的，前端连点、消息重发这类场景可以放心。

四、M3 多人并发评价同一活动（信用流水核对表）

设计：被评人起始 50 分，15 名参与者并发各评一次（每评 +2，封顶 100 不触发）。
期望：15 条流水、score_after 逐笔 +2 递增（52→80）、终值 80。

实测（多轮形态一致，以最后一轮为准）：

核对项	期望	实测
评价成功数	15	15 ✓
库中评价行 / 流水行	15 / 15	15 / 15 ✓
score_after 序列	52,54,56,...,80 严格递增	**52×10, 54×5** ✗
终值	80	**54** ✗
丢失更新	无	**有：+30 只到账 +4**

BUG-7（P1）：并发评价信用分丢失更新。`EvaluationServiceImpl` 加分是
"SELECT 读分数 → 内存算 52 → UPDATE 整值写回"，并发事务读到同一旧值后互相覆盖，
流水却每笔都记了账 —— 档案与流水对不上账，15 人评价 30 分的信用增量蒸发 26 分。
修复建议：改 `UPDATE user_profile SET credit_score = credit_score + 2 WHERE user_id = ?`
（原子自增，封顶用 `LEAST(100, ...)`），或 `SELECT ... FOR UPDATE`。
守门用例 `test_cc02`（xfail 挂账，5 人并发 +10 必须精确到 60）。

五、安全用例（tests/test_07_security.py，10 条）

组	用例	结果
认证	SEC-A01 无 token / A02 垃圾 token / A03 改 payload 不改签名	全 401 ✓
越权	SEC-B01 水平越权退他人报名（4040，本人可退）	✓

组	用例	结果
越权	SEC-B02 待核验发起活动 4019，报名不受限	✓（实名闸门符合三档设计）
实名	SEC-C01 名册命中免造数直发活动 / C02 冒用学号 4018 且文案不泄露归属	✓（走演示名册）
注入	SEC-D01 登录 SQL 注入 4 组 payload / D02 存储型注入原样入库、表完好	✓（参数化语句有效）
密钥	SEC-D03 用仓库默认 dev secret 离线伪造 userId=1 的合法 token	**伪造被接受** → 挂账

BUG-8（P2，配置面）：`bugedu.jwt.secret` 默认值 `buge-dazi-dev-secret-change-me` 随仓库可见，本地未设 JWT_SECRET 环境变量时，任何拿到仓库的人可离线签发任意 userId 的合法 token 冒充任意用户（签名校验本身拦不住"合法签名的冒用"）。
修复建议：部署侧强制注入随机 JWT_SECRET 并废弃默认值；测试守门 `test_sec_d03`（xfail 挂账）。

六、BUG-4 / BUG-5 结案确认

李佳 `e09ce85`（互评资格校验）合入后，Day4 两条挂账按预期转正：

Day4 挂账	验证	处理
BUG-4 未校验参加关系/结束状态	越权评价 → 4016；未结束评价 → 4017	xfail 转硬断言（回归守门）
BUG-5 被评价人不存在 → 5000	资格层在 insert 前拦截 → 4016，5000 路径消除	按新契约 4016 断言，结案

七、环境变更记录（复现须知）

1. master 新增依赖：本地库需跑 `sql/day5_real_name_roster.sql`、`day5_verify_status_migrate.sql`、`day6_sensitive_hash.sql`（幂等）；真实名册 216 人已导入 `real_name_roster`（数据文件在 `sql/local/`，gitignore 保护，不入库不外传）。
2. 本地 application.yml 需补 `sensitive.crypto.aes-key-base64`（32 字节 Base64）与 `pepper`，
   否则后端起不来（SensitiveCryptoConfig 强制要求）。
3. 测试造数学号改为 9 位纯数字（`data_factory.seed_student_id`）；注册即待核验，
   flows 对发起人账号做库侧核验造数（`flows.verify_user`，与 set_credit 同性质）。
4. 消息分页契约变化：size/page 越界从"静默收敛"改为显式 4001（林严泽 ddb9b20），
   BD-M01/M02、BD-P04（4040→4030）已按新契约更新。

八、明日（Day6）建议

- 把 BUG-6/BUG-7 的修复排给李佳（都是事务内改动，Day6 上云前修完风险最低）；
- BUG-8 在 Day6 上云时一并处理（云端部署本来就要注入 JWT_SECRET）；

- 修完即跑 `perf/mixed_load.py --scene ALL` + `test_08_concurrency.py` 验收，去掉两条 xfail 标记转正式回归。