服务器容灾测试说明

——测试部测试中心

1. **测试概述**

**1.1、服务器容灾测试**

服务器容灾测试，是指测试人员、运维人员，针对趋于成熟的服务器架构，以人为损伤的方式在单点程序崩溃，系统过载崩溃，系统宕机，数据存储异常，程序服务性能及效率等方面的综合情况测试，以验证被测架构的合理性，健壮性

**1.2、测试目标**

验证，优化，完善服务器架构，为公司项目（代理项目）在运营可靠性评估，架构资源分配等方面做抉择参考

**1.3、测试周期**

周期：4~6日/人，周期内细则安排如下

需求沟通和方案设计：1~2日/人，环境搭建：1日/人；

一轮完整的测试并产出报告2~3日/人。

**1.4、测试条件**

1) 服务器不再做新的功能，容灾物理资源申请到位

2) 部署必须和正式发布环境一样，集群的节点必须有2台以上，主备机都有

3) 开发商提供：容灾容错设计检查表（选）、架构设计文档（必）、环境说明文档（必）、连接容灾环境的PC版本（必）、可供容灾测试的服务器搭建文件

1. **测试策略**

**2.1、依据容灾标准，进行服务器/客户端等的人为损伤模拟测试，其结果做为容灾结果评估**

1. **测试指标评定标准**

**3.1、容灾基线**

低风险，无遗留严重致命缺陷为服务器容灾接受基线

**3.2、测试标准**

见附录1说明《测试项及评定标准（服务器容灾）》

见附录2说明《测试项及评定标准（服务器容错）》

**3.3、评定标准**

以服务器容灾容错测试项标准做为执行过程向导和测试结果参考，结合容灾基线，综合为服务器容灾测试的评定标准

1. **测试终止标准**

**4.1、环境终止**

当前提交的版本出现影响测试的重大问题情况下，将终止本次提交的版本测试，包含但不限于：服务器环境/架构异常，游戏主功能，主流程，主UI异常等

**4.2、过程终止**

测试过程中，版本出现突然事件处理需要更换版本的，终止当前版本容灾测试

测试过程中，出现项目优先级，项目终止、滞后等第三方原因，终止当前版本容灾测试

**4.3、结果终止**

完成本次容灾测试相应测试领域并做好数据统计，结果梳理，终止本次容灾测试

1. **结果输出**

**5.1、测试文档**

终止测试后，测试将产出《容灾容错检查测试用例》，《容灾容错测试报告》

**5.2、结果定义**

**高风险：**

以下任意情况发生，其结果即为高风险：

1. 单点风险：单个程序失效导致玩家超过15分钟无法游戏
2. 过载风险：单个系统可能过载，并导致所有玩家响应变慢或者无法使用
3. 意外停服风险：故障切换时需要停服重启以载入新配置
4. 长时间回档风险：因数据变更未及时入库，导致可能发生10分钟以上的回档
5. 重要数据异常风险：任何可能导致玩家重要数据（例如货币、高价值物品）错误的风险

**低风险：**

以下风险如果策划和运维方已知晓并接受，则可通过：

1. 分布式架构中，单个程序或机器故障无法自动切换，需要人工完成故障切换
2. Cache延迟写方案中，意外故障可能导致不超过10分钟的数据回档

**完备：**

不存在任何风险，各系统有全面的风险应对和故障自动切换措施

1. **附录**

**附录1：****测试项及评定标准（服务器容灾）**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 地域 | 备份 | 类型 | RPO | RTO |
| 接入层 | 异地 | 冷备 | 应用容灾 | / | 5分钟 |
| 逻辑层 | 同IDC | 磁盘阵列 | 应用容灾 | 分数、二级货币、属性值等数据不影响超过0.1%的用户，且不能减少超过玩家10分钟内获得的所得。一级货币不允许对视 | 玩家超过15分钟无法体验核心游戏玩法 |
| 数据层 | 同IDC | 同步备份 | 应用容灾 | 分数、二级货币、属性值等数据不能影响超过0.1%的用户，且不能减少超过玩家10分钟内能获得的所得。一级货币不允许丢失 | 玩家超过15分钟无法体验核心游戏玩法 |

**附录2：测试项及评定标准（服务器容错）**

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 级别 | 检查项 | 描述 |
| 1 | 高 | 客户端过载保护：连续三次登陆失败禁止一段时间登陆 | 登陆界面登陆失败时，不允许程序自动重新登陆，需用户自己点击重试。连续3次登录失败后，提示用户等待10秒后才能登陆，登陆功能暂时不可用，再连续3次失败后需要等待5分钟以上。客户端重启也需要记住间隔时间 |
| 2 | 高 | 客户端模块隔离：单个后台服务的异常不能导致客户端不相关逻辑不可用 | 单个后台服务全部进程停止、无法连接、连接超时、返回错误码时，不能影响用户使用不相关的其他功能，例如：支付接口异常，不应该导致客户端死循环不能登录，社交服挂掉，不能影响战斗服的进行 |
| 3 | 高 | 服务器：不能有单点故障或单点故障时间少于15分钟影响 | 每个服务都有多个节点进行应用容错（根据用户账号固定归属节点只有1个也算单点），单个服务进程异常结束、单个服务器异常关机掉电不能让用户感知。如果有单点故障必须保证服务中断时间少于15分钟，不能影响超过5%的用户登录或体验游戏核心玩法 |
| 4 | 高 | 服务器：单个服务器进程异常结束后不能导致用户关键数据丢失 | 数据库服务异常结束后不能扣费不发货；内存缓存脏数据不能是关键用户数据，若脏数据回写时间内数据丢失：分数二级货币属性值等数据不能影响超过0.1%的用户且不能减少超过玩家十分钟内的所得。一级货币不允许丢失。玩家重要数据（例如货币，高价值物品）必须写运营日志 |
| 5 | 高 | 服务器：单个硬件永久损坏不能导致用户数据丢失 | 永久性存储需要有冗余备份 |
| 6 | 高 | 服务器模块隔离：单个服务支持独立重启 | 不能与其他服务有启动顺序的依赖关系 |
| 7 | 高 | 服务器模块隔离：单个服务器进程异常结束不能引发其他服务进程异常 | 每个服务调用其他服务都得有容错保护，当错误发生时，特别是在高负载下，不能引起雪崩 |
| 8 | 高 | 服务器模块隔离：外部接口异常不能影响用户关键路径 | 接口无法无法连接、调用超时，玩家可以正常执行与之无关的操作 |
| 9 | 高 | 服务器平滑扩（缩）容：可在运营中动态调整服务节点个数 | 可以在任意时间新增 或者减少服务节点（进程或者机器），不能产生用户掉线和数据不一致。Gamesvr通常应该设计为无状态服务，支持运行时平行扩容。Cache和DB程序都应该具备2小时内扩容100万PCU的能力。DB设计要支持分布和扩容，即允许通过配置修改DB路由，不可在Gamesvr中写死DB数据和路由 |
| 10 | 高 | 服务器过载保护：接入层需要有阀值控制 | 有阀值控制最大同时在线、每秒新建连接数、单个连接每秒发送请求数等，超过阀值就拒绝连接 |
| 11 | 高 | 服务器过载保护：服务器异常对用户下线需要返回错误码 | 游戏服务器主动关闭用户连接时，需要返回错误码后再断开连接 |
| 12 | 高 | 服务器过载保护：服务器对一段时间内不活跃的连接要强制断开 | 防止拒绝服务攻击 |
| 13 | 中 | 服务器模块隔离：游戏服务进程与接入层进程剥离 | 游戏服务进程重启（升级）不影响已经连接上的用户（重启gamesvr不影响玩家已经跟接入层建立的tcp连接）不需要用户进入重新登录界面，让用户无感知 |
| 14 | 中 | 服务器：服务启动耗时小于30秒且不能影响用户体验 | 包括第一次启动和重启，共享内存数据加载。重启过程中不能影响用户体验，如：掉线，超时等待，卡死等。仅对需要在线重启且不支持路由自动切换的程序有要求 |
| 15 | 中 | 服务器负载均衡：同等服务和数据是动态负载均衡的 | 接入层、逻辑层和存储层的用户请求和数据会在不同的节点之间动态均衡，机器或者服务的负载一致 |
| 16 | 中 | 服务器：服务器支持进程消失后自动拉起 | 1分钟内能检测到并自动拉起。重启服务器也需要检测并自动拉起 |
| 17 | 中 | 服务器：关键进程和性能计数器需要接入监控告警 | Zabbix监控 |
| 18 | 中 | 不能 意外停服 | 不能因为客户端功能缺陷，导致服务器需要重启或者停服。客户端版本升级不能导致服务器需要重启或者停服。服务器需要支持强制升级和选择升级。服务器要支持动态加载配置，防止需要停服重启以载入新配置 |