工程师

对于数据库恶意删除，可分为员工主动恶意删除和攻击者恶意删除。

1. 数据库服务云上灾备（防攻击者或其他异常）
    - 方法：将数据库服务部署在两个云服务商上，每个云中运行两个实例互为温备份，两个云的主实例互为热备份且双活。在存储系统外使用负载均衡服务控制不同云的实例连接情况，总架构为双主双从。每周对整个数据库系统做增量冷备份，保存在公司内独立存储服务器上。
    - 花费计算：以AWS为例，单个云上2个实例，实例配置为4个vCPU，32GiB内存，10Gbit最大网络吞吐量，每月730小时运行时间，开启RDS代理，数据库容量2TB， 基线每秒IO30次，峰值IO100次，峰值活动时长大约30%，额外备份存储1TB。每月总存储成本244.99USD，额外存储成本23.55USD，数据库实例1005.94USD，RDS代理成本128.48USD。每月数据库服务总成本2871.92USD。外层接入Cloudflare保活，每月200USD。总数据库系统每月成本3071.92USD。
      本地备份受存储容量动态影响，以西部数据HC550 18TB SAS接口硬盘为例，每TB存储价格为150元，存储服务器价格不超过5万元，以RAID5方式构建阵列，四个硬盘为一组存储池，平均每TB存储价格为200元。单次数据备份系统的搭建花费为53000元左右。
2. 访问权限控制（防员工）
    - 用户验证：启用双因子认证（两步验证），第一步使用静态口令，第二步使用基于TOTP（Time-Based One-Time Password）的虚拟验证器。对于关键高权限用户（例如存储系统管理员、安全管理员、数据库操作审计员等），启用生物特征识别，例如使用指纹验证。
    - 控制数据存取权限：数据库系统中使用 `GRANT` 命令授权，使用 `REVOKE` 收回权限，使用 `CREATE ROLE` 和 `CREATE USER` 创建角色和用户。
    - 数据安全标记：将不同数据按敏感度分级，以行为最小粒度分级。同时用户有对应不同分级的许可，低许可写入高密级，高许可读取低密级。并使用视图（View）机制将需要控制访问的数据对无权访问的用户隐藏。该权限隔离机制需要经过形式化验证。
3. 数据库故障恢复（应对方案）
    - 使用检查点（Checkpoint）机制和Postgresql自带的多版本并发控制（MVCC）：实现数据库自身的故障恢复。要求复杂业务（例如对数据库有批量插入或删除等操作）在提交修改前后创建检查点。
    - 设置连续归档和基于时间点的恢复（PITR）：使用pg_basebackup创建基础归档并配置连续归档功能。当出现误操作后，查找对应WAL（write-ahead-log）日志编号，关闭数据库服务并通过指定事务号的方式重置数据库（pg_resetwal）。若需要基于时间点的恢复，则需要在关闭数据库后修改配置文件（postgresql.conf）并编写恢复相关参数（如restore_command），之后启动数据库恢复模式进行数据恢复。该方法支持时间戳、检查点和事务ID的恢复目标，且恢复数据也依赖WAL日志。
    - 基于触发器的闪回：在敏感表中添加标记行，并设定触发器函数监测标记行是否存在。当标记行被意外删除时，触发报错并终止操作，回滚事务。
    - 容灾策略：当一个云服务宕机时，负载均衡系统会直接切换至另一个云服务并产生告警。此时立刻由安全管理系统终止对业务系统的更新和修改，对外执行业务降级方案。人工校验本地备份与云服务的温备份，根据备份情况在第三个云服务商上开起临时实例，以热备份模式运行，总架构变更为一主带两从，恢复正常业务等级。直到原服务商恢复服务后，将主服务恢复至原先的热备份实例，同步后重新运行，关闭临时实例，总架构恢复为两主带两从。

首席技术官

拥有更高权限，可修改公司的制度和架构来进一步保证数据库安全。

1. 安全管理系统
    - 审计功能：启用pgaudit扩展，以会话（session）模式监控以下操作（位于pgaudit.log中）：READ、WRITE、FUNCTION、ROLE、DDL、MISC、MISC_SET、ALL。操作审计员需要确认对数据库的关键更改，校验命令内容、命令执行人、执行时间、命令传输路径、命令发送机器，并留存日志。
    - 安全管理员：
        - 维护可读的安全列表，其中包含被控制的对象和访问权限说明。
        - 负责维护“可访问机器”。每个用户仅在公司内的指定机器上允许操作数据库系统，若不在指定机器上操作，则任何用户在公司内其他机器上仅能以最低许可访问数据库。特殊的，可访问机器不允许连接其余外网地址，并位于内网独立隔离网段，使用内网的防火墙控制访问策略。安全管理员需要维护“可访问机器”的机密性、完整性、可用性。这些“可访问机器”的任何操作记录均会被完整保存，系统支持回放操作和还原点功能，有独立硬件支持的数据加解密芯片作为可信基，利用该可信基完成数据传输全链路加密。
        - 在任何用户操作前，管理员需要验证每个操作的权限，并将试图访问的过程记入日志中。若用户主动取消访问，仍会被记入日志中。此过程的部分日志可能由数据库以外的其他设备记录，如上网行为管理、内网流量探针、内网防火墙、数据库应用防火墙等。
    - 部署更多网络安全设备：例如入侵检测系统，检测网络流量中是否存在高危数据库命令执行。
2. 项目开发规范
    - 网站应用开发：要求每个业务块以事务形式提交，使用可重复读策略避免数据异常。对数据库的写操作使用存储过程封装，不在网站应用对外信息传输中加入SQL命令，所有对数据库的操作由网站应用内代码直接发起。
    - 代码审查：在代码编写后，由研发经理的团队进行代码审查。要求公司内有代码规范和标准文档，关注整体设计、代码功能、代码复杂度和其他开发规范。开发人员需要以注释或其他规定形式详细说明代码意图，审查人员需要针对每部分代码攥写评论。当一个完整模块或规定代码量被审查后，需要由代码审计员留存日志后归档。
    - 代码测试：在代码测试阶段，测试组人员应当与开发组人员隔离（指不能同时有人既为测试员又为开发员），且测试组不具有修改项目代码的权限。每次测试结果应当由测试审计员留存日志，防止隐瞒测试结果。测试组应当按照要求如实发布测试报告并反馈给开发组。
    - 项目发布：在项目发布阶段，使用灰度发布策略。不直接替换生产环境的旧版本，而是先另启动少量新版本应用实例与对应的后端数据库独立实例，在业务负载均衡系统中少量切换原有流量至新版本。在新版本应用运行时，测试人员持续收集运行数据和业务日志，对比两版本差异，且新版本应用的数据备份频率提高至每日一次。确认新版本应用运行正常后，再完全切换流量至新版本应用，回收旧版本所有资源，完成灰度发布。当发布过程中出现异常时，立即将流量全部切回旧版本应用，并根据备份和业务日志恢复数据，保证业务正常运行。
3. 系统安全杂项
    - 更新补丁与监控：数据库和其他业务组件均应定期安装官方补丁，及时更新最新版本。由数据库安全管理员监控数据库活动的数据，包括性能和实时操作。设置告警规则以快速响应威胁。
    - 设置严格的密码策略：要求公司员工定期修改密码，且密码强度要足够复杂，例如在KeePass Estimation中达到中等强度，高权限用户需要达到强密码。
    - 应急方案：公司应当对所有威胁事件有准备应对方案并形成文档和自动化工具，培训每个岗位的员工知晓自己负责的模块应当做何种应急操作。同时也应培训员工的数据安全意识和网络安全意识。