完整性

一.完整性需求

1.商业系统的完整性需求

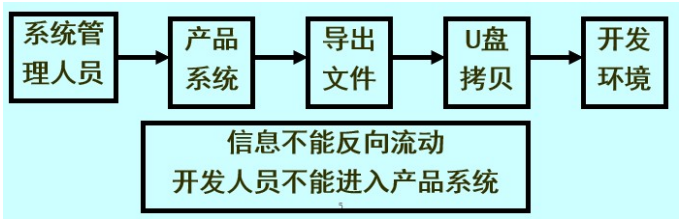
- 如果用户自己编写程序、修改、使用数据，系统能否正常运行？

  不能，**应使用产品程序、数据库**

- 对开发人员有何需求？
  - 能否在在线运行的系统上调试程序、修改数据？

    不能，**建立开发环境，在开发环境中模拟、测试**

  - 在开发环境中需要使用产品数据，如何使用？能否登录进入产品系统中，直接取出数据？

  - 产品开发周期结束，上线运行前需做哪些准备？

    割接过程,**需专门的处理过程完成转换工作**

    > *如：银行的新系统上线通知，用户不能使用系统*

  - 割接过程仅需开发人员？如果过程中出错，怎么办?如何知道出错与否？

    需被控制、审计

  - 系统运行过程，不保留交易信息如何？

    管理员查看系统状态

    审计人员查看审计日志

2.通用操作规则

1)责任分离

- 需要多步完成一个事务时，需2个以上人员共同完成

2)功能分离

- 开发环境、开发系统 与 产品环境、产品系统分开
- 开发人员不能在产品数据上做操作

3)审计需求

- 商业系统：保证可恢复、问责、能够记录日志、可审计

  审计人员：进行审计

  特别地，从测试环境进入产品环境，需记录日志、审计

二. Biba模型

1.内容

- 系统中，包含主体集S，客体集O

  主体和客体具有完整性级别 I，包括完整级和分类

  > I_1={完整级1，{哈市市行数据} }
  >
  > I_2={完整级2，{哈市市行数据，牡丹江市行数据} }

- 完整性级别的比较

  关系$I_1 \leq\subseteq I_2$成立：当I_2 dominate I_1

- 完整性级别越高，越信任其准确性

  主体的完整性级别高，信任其能够正确写入

  客体的完整性级别高，信任其是正确数据

2.完整性模型

- 和BLP模型相反
- 1)读操作

  s∈S can read o∈O iff i(s)≤i(o)

  2)写操作

  s∈S can read o∈O iff i(o)≤i(s)

  3)执行操作

  s_1 ∈S can read o∈O iff i(s)≤i(o)

  4)既能读又能写

  i(s)=i(o)

三.Clark-Wilson模型

- 模型考虑如下几点：

  1）主体必须被识别和认证

  2）客体只能通过规定的程序进行操作

  3）主体只能执行规定的程序

  4）必须维护正确的审计日志

  5）系统必须被证明能够正确工作

1.实体

- CDI: 受限数据项数据要受完整性约束
- UDI: 非受限的数据项数据不受完整性约束
- TP: 交易过程
    - 将系统从一个有效状态转移到另一个有效状态的过程
- IVP: 完整性验证过程
    - 检查CDI遵守完整性限制的过程

2.证明规则

- Certification rule (CR1)

  当任意 IVP 运行时, 它必须保证所有的CDI处于有效状态
- Certification rule (CR2)

  对相关联的CDI, 一个TP必须将这些CDI从一个有效状态转到另一个有效状态

  一个特定的TP和几个相关CDIs相关联

  > *例: TP是存取程序, CDI是存取前后的钱数*

- Certification rule (CR3)

  系统执行操作时，符合责任分离原则

3.实施规则

- TP-->CDI ：保证具有关联关系
- Enforcement rule（ER1）

  系统要维护关联关系，保证经过验证的TP操作相应的CDI
- Enforcement rule（ER2）

  TP操作CDI时，保证操作用户有权对相应CDI做操作，TP所代表的用户是CDI的真实用户

  三元组 **{ user, TP，{CDI set} }**
- Enforcement rule（ER3）

  系统执行TP前，应验证用户身份

  > *验证客户身份，登录系统的操作员身份*

- Enforcement rule（ER4）

  只有可以授予TP访问规则的主体才能修改列表中相应的表项，授权主体不能执行TP操作