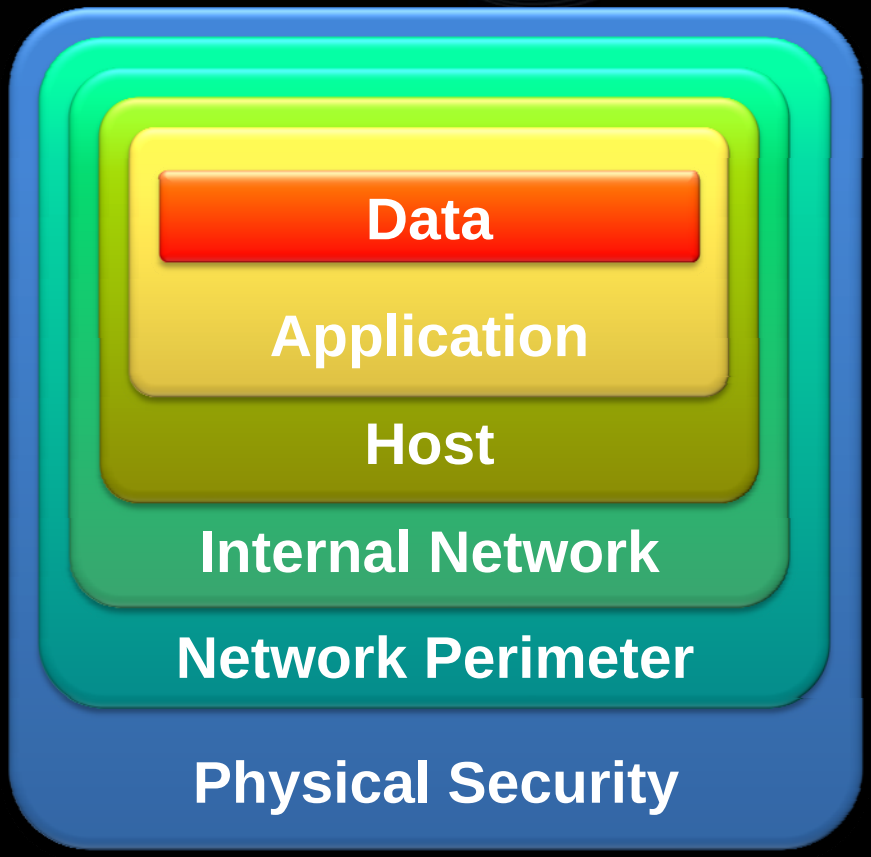
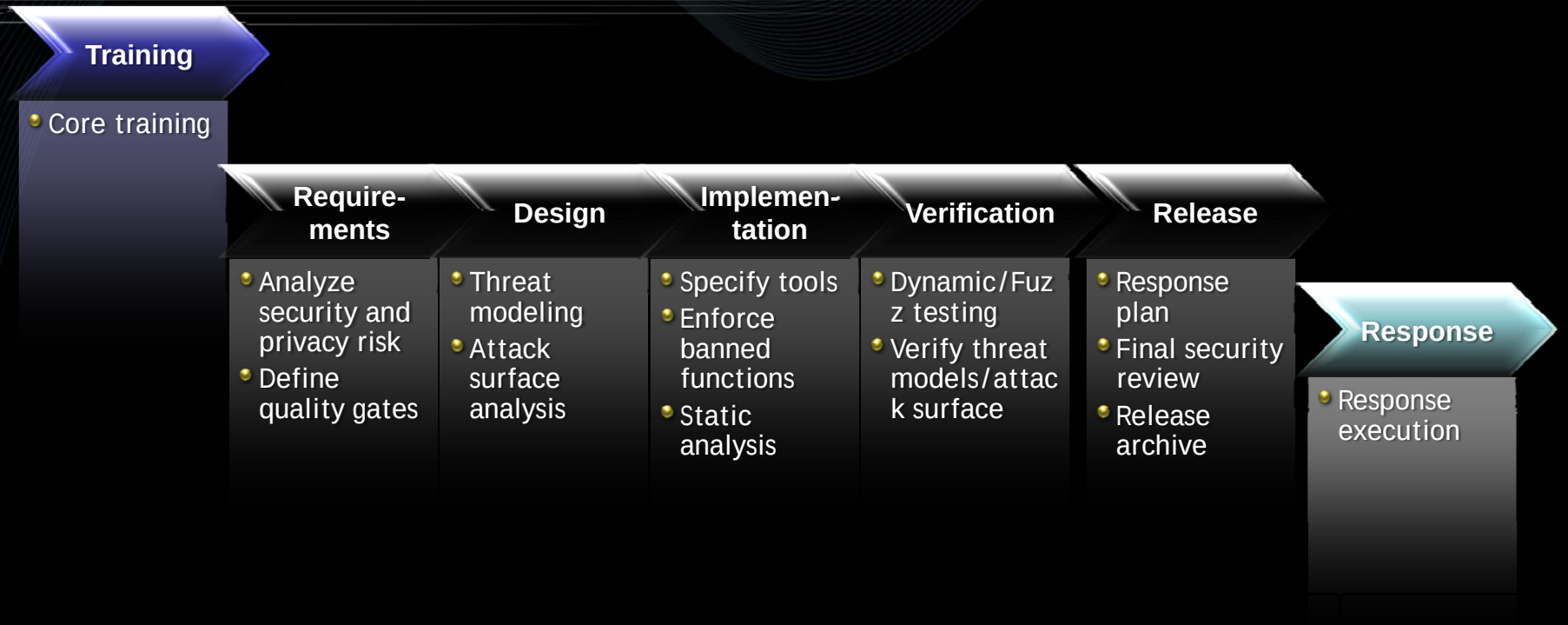
基于数据流SDL的安全测试

南京翰海源信息技术有限公司
FlashSky

多层保护体系->保护越来越向核心迁移



SDL模型



Education Technology and Process Accountability

过程改进是渐进的，并且不需要在发展进程中彻底改变。但是，重要的是使整个组织持续改善。

SDL开发过程环节与关联性

- 需求分析
- 概要设计
- 详细设计：？
- 编码
- 测试

开发过程连贯性

- 概要设计:按照攻击界面威胁建模
- 详细设计：？
- 编码：按照经验和规约实施和检测
- 测试：按照经验和功能点进行黑盒FUZZ 测试

SDL开发过程工具支持

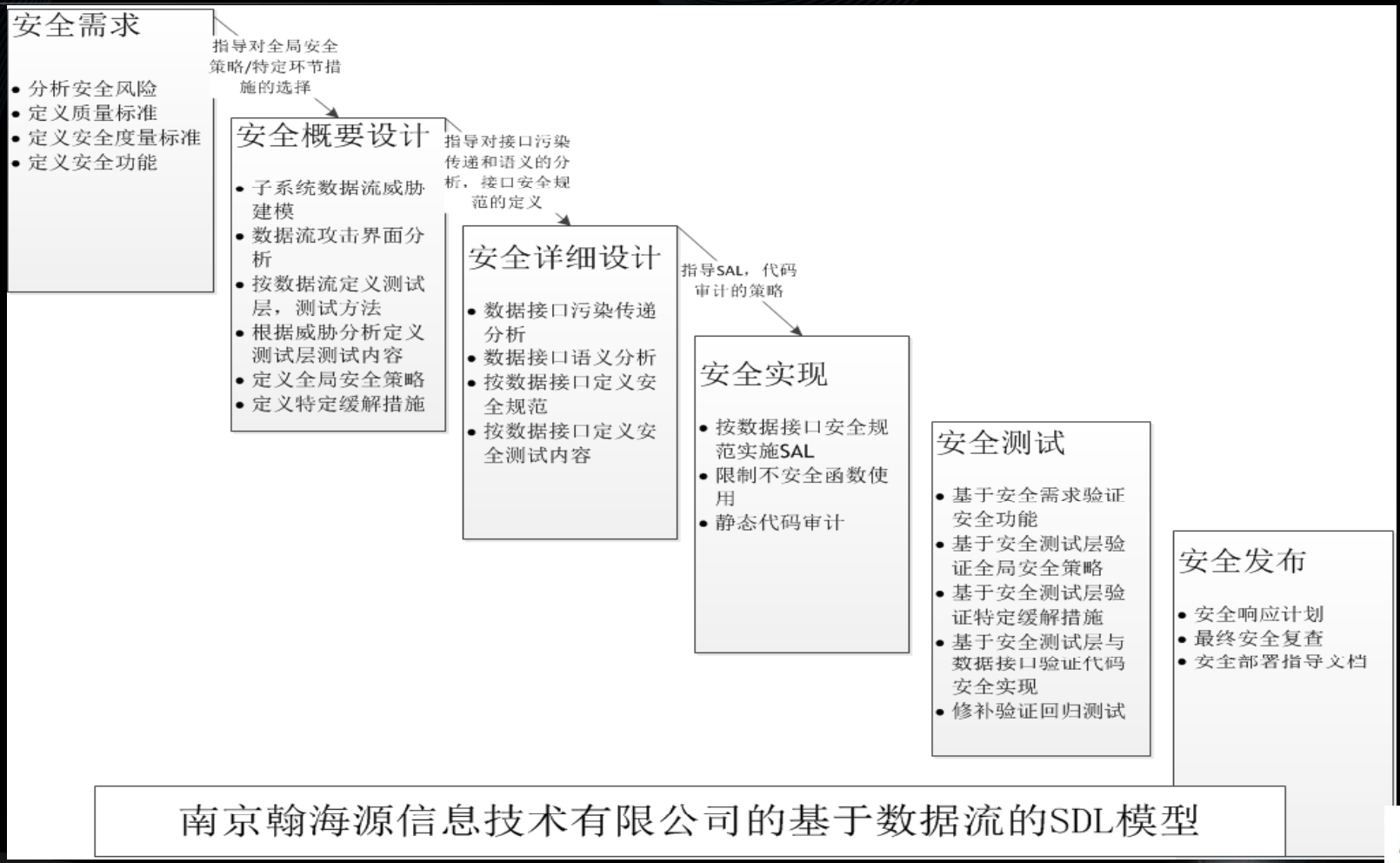
- 概要设计:VISIO
- 详细设计：？
- 编码：代码审计-有
- 测试：缺乏通用的产品和工具

- 整体管理支撑：无

可剪裁的SDL过程

- 企业实施的可行性
- 投入与风险
  - 发现错误的成本：前提条件是团队已具备安全意识和能力
  - 软件工程能力度
  - 改造过程和团队的成本/风险
- 安全本身就是成本度量
  - 没有绝对的安全
  - 攻击者成本>攻击者收益>>安全成本就是安全
- 可裁剪逆向的SDL
  - 逐步渐进可裁剪的SDL
  - 风险可控

基于数据流的SDL模型



安全需求分析->标准度量

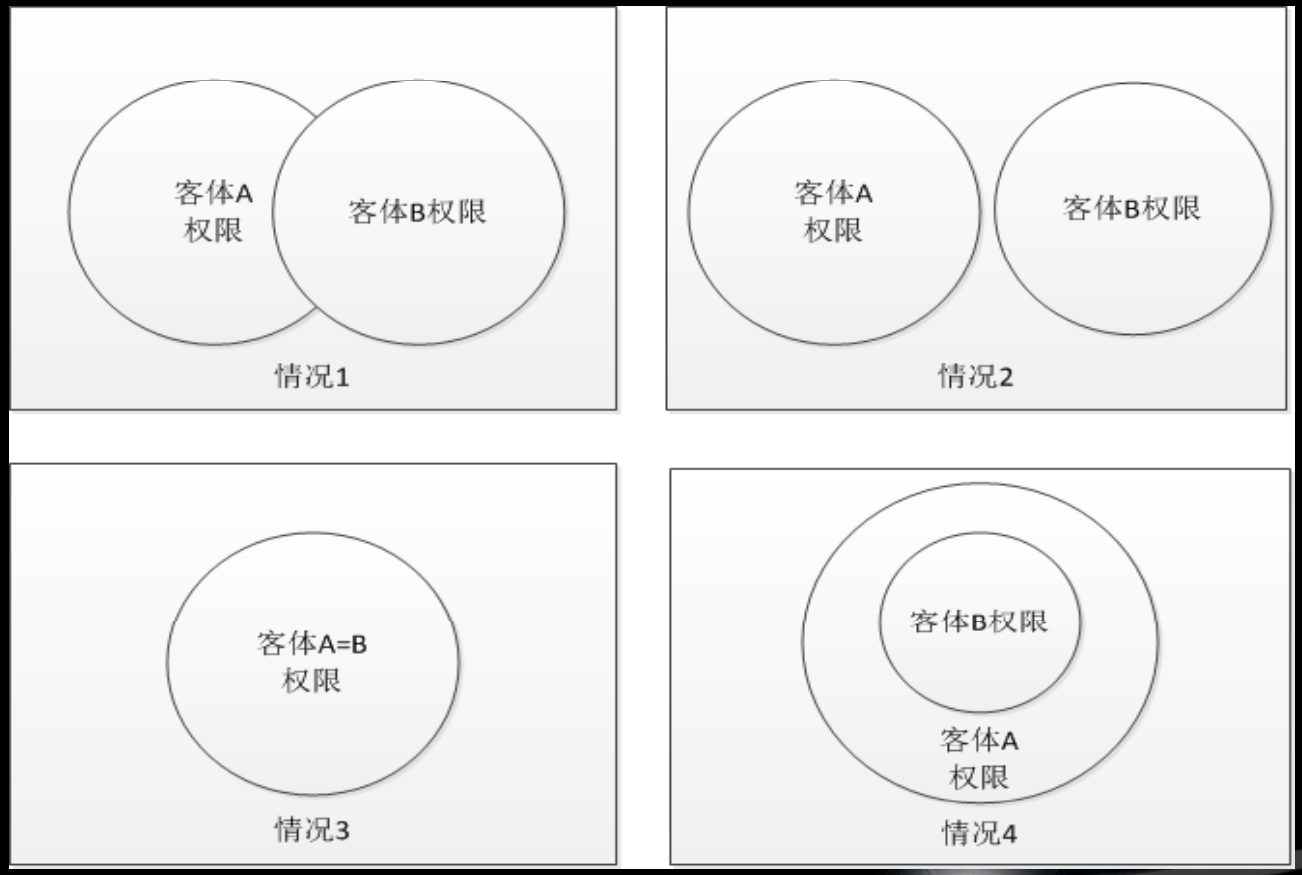
- 如何估量攻击者成本>攻击者收益>>安全成本？
  - 保护资产/安全风险/成本分析
- 如何规定安全指标
  - 安全功能设计
  - 全局安全策略原则
  - 安全实现原则和测试覆盖性和度量性指标
- 如何验证和度量安全
  - 安全验证
  - 安全测试
  - 安全测试覆盖性
  - 安全测试度量性分析

安全问题核心

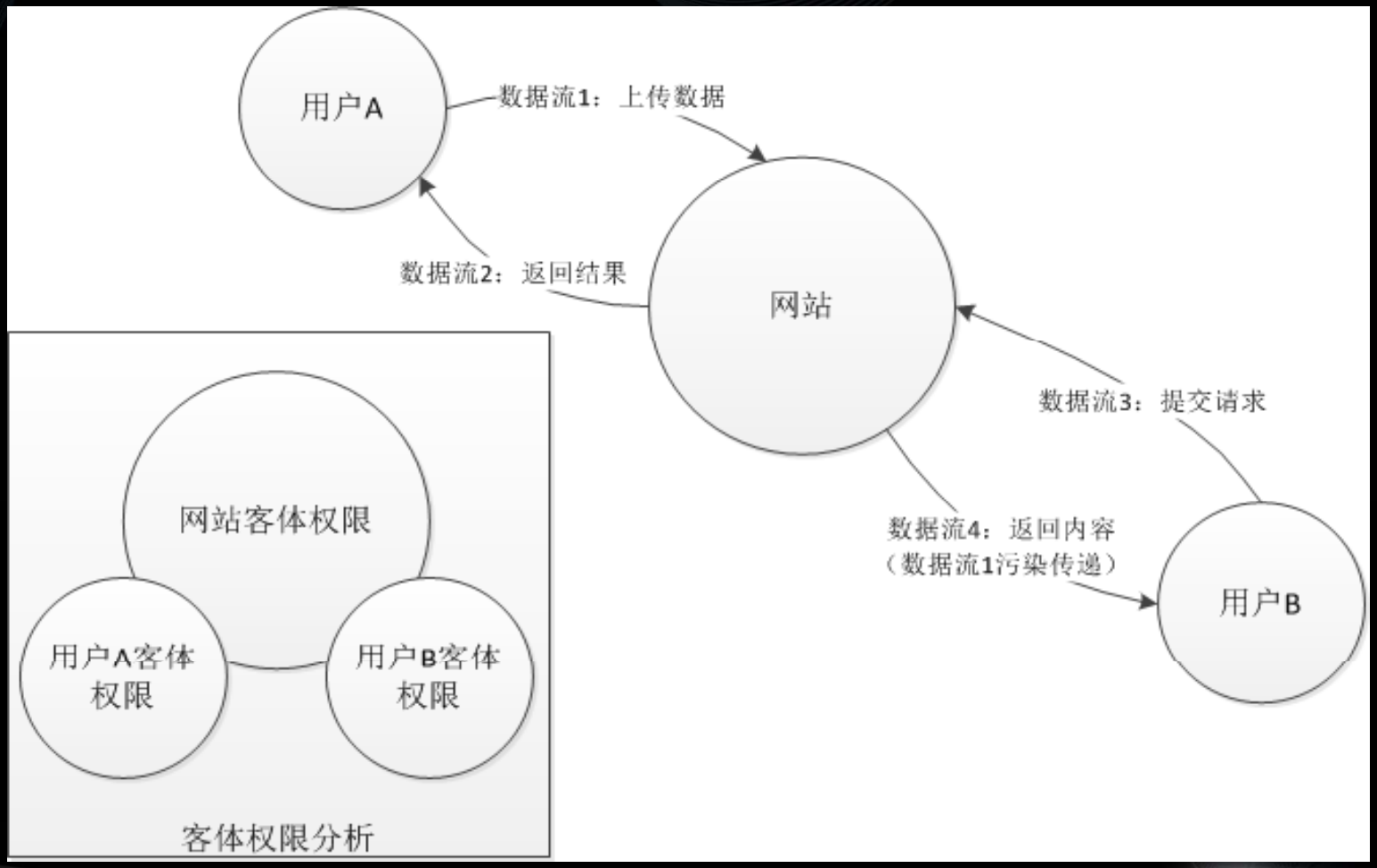
- 安全问题的本质是
  - 系统代表A用户身份
  - 处理B用户可控的数据内容或操作
  - 系统对数据内容或操作的处理存在安全漏洞

威胁建模->攻击界面分析

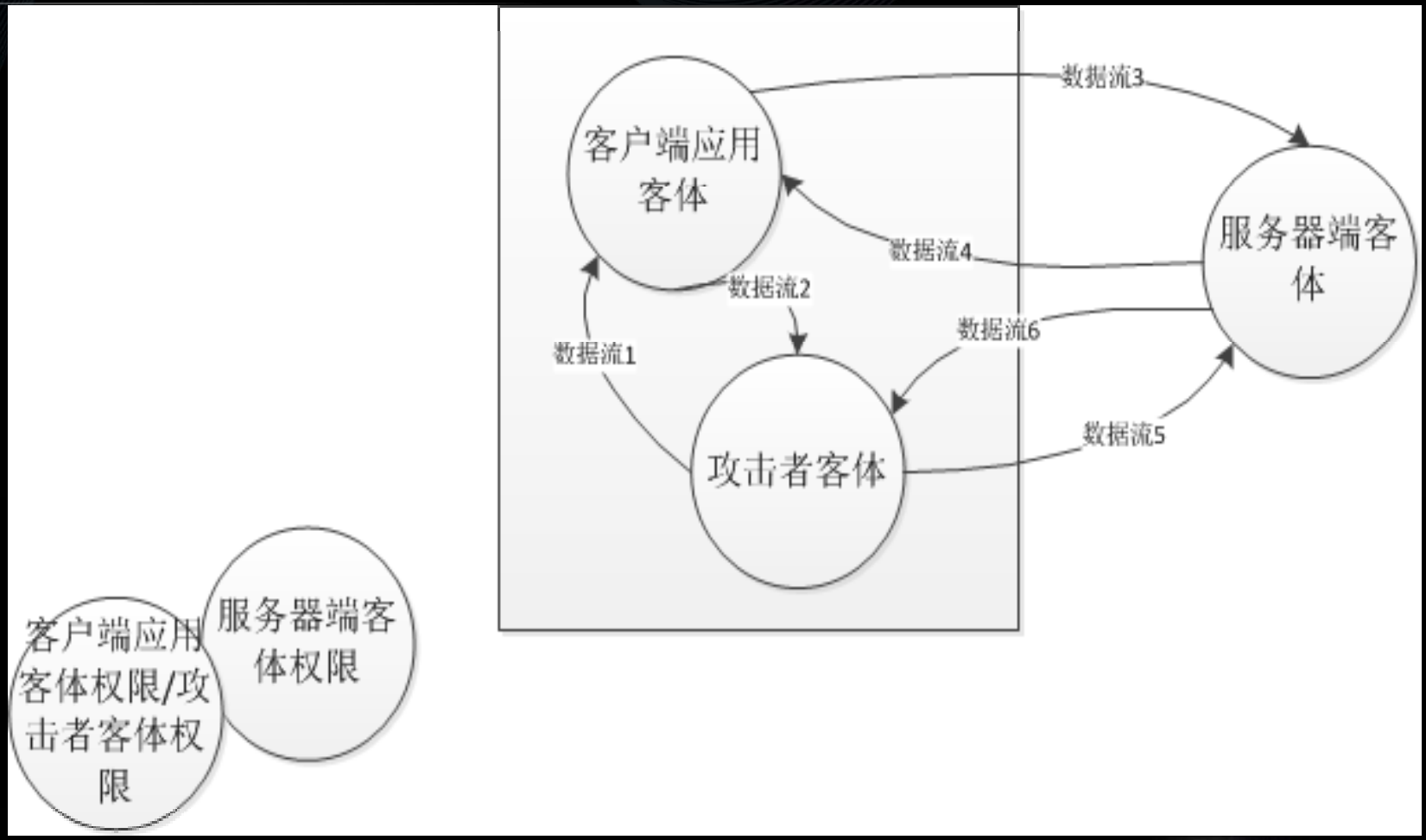
- 不同权限客体之间的数据流就是攻击界面
  - 客体
  - 数据通道
  - 数据流
  - 权限
  - 双向性



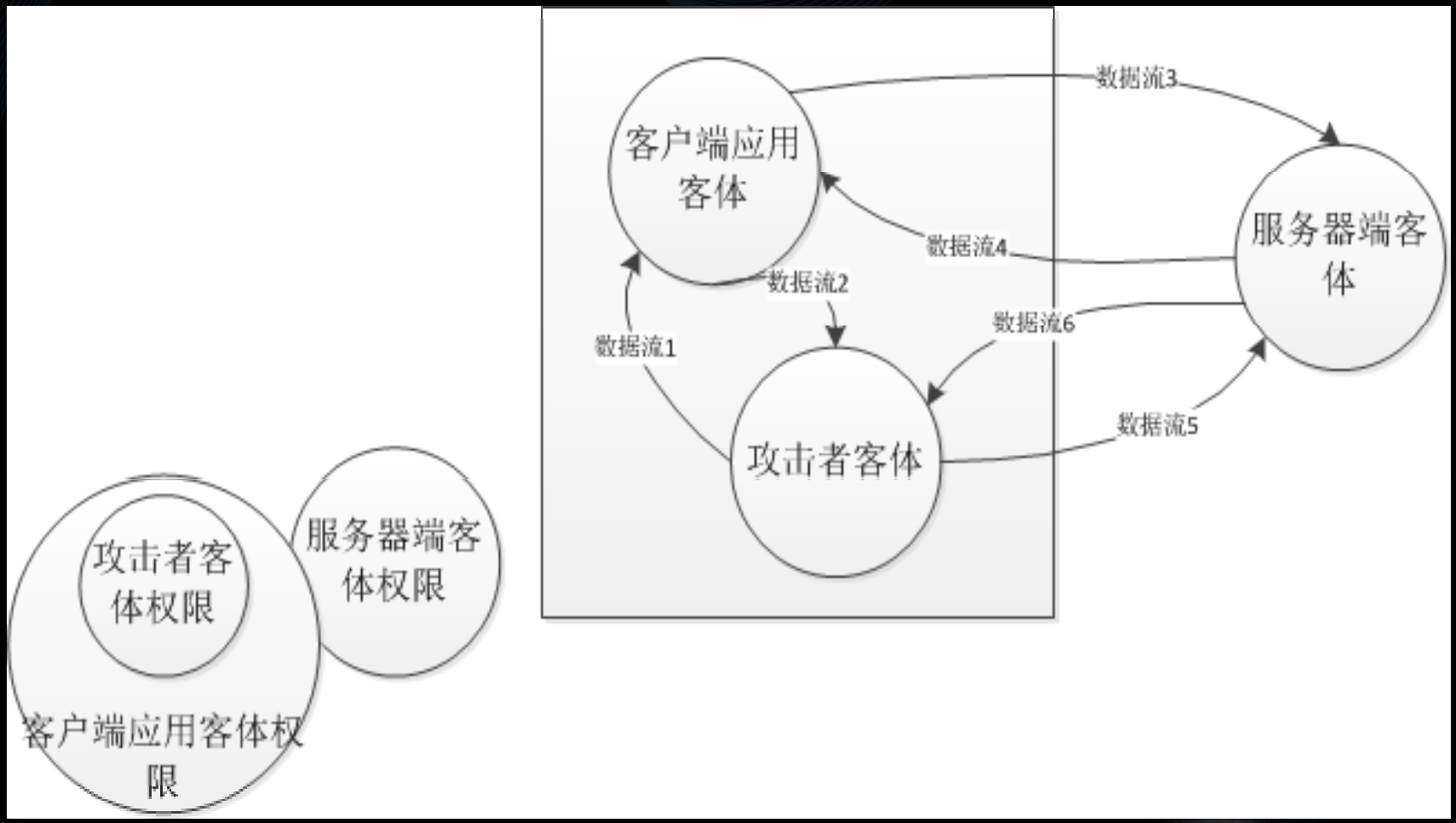
威胁建模->攻击界面分析实例



威胁建模->攻击界面分析实例



威胁建模->攻击界面分析实例



基于数据流的威胁分析

- 数据如何被使用的
  - 存储
    - 数据库
    - 文件
  - 操作
    - SQL语句
    - 内存处理
  - 显示
    - 混合内容
  - 传递

全局安全策略分析

- 部署性安全策略
- 配置性安全策略
- 安全设计体系性安全策略

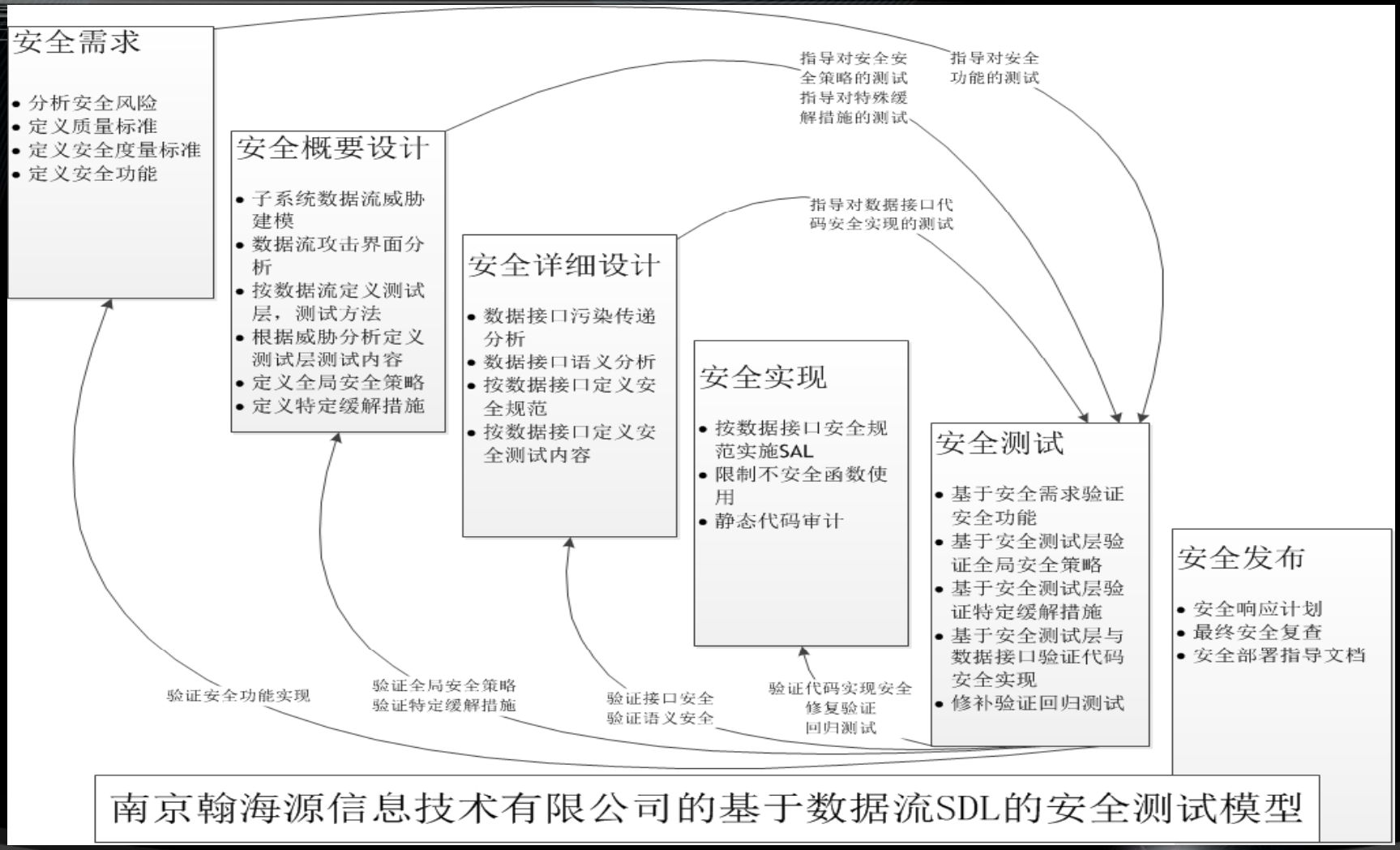
安全详细设计

- 污染数据接口传递分析
  - 清晰知道可能受影响的区域和问题
    - 指导安全编码
    - 指导代码审计
- 缓解措施定义
  - 具体的缓解措施和缓解类型

定义安全测试

- 安全包括了三个层次
  - 安全功能（特性）
  - 安全策略（部署，配置，全局设计准则）
  - 安全实现
- 安全测试是对以上几个层次的验证和度量

基于数据流SDL的安全测试模型



安全功能实现验证测试

- 验证
  - 定义的安全功能是否正确实现
  - 定义的安全功能是否符合默认配置
  - 定义的安全功能是否全局有效
  - 定义的安全功能实现的强度和访问权限保护
  - 定义的安全功能是否可以被绕过（自动降级）

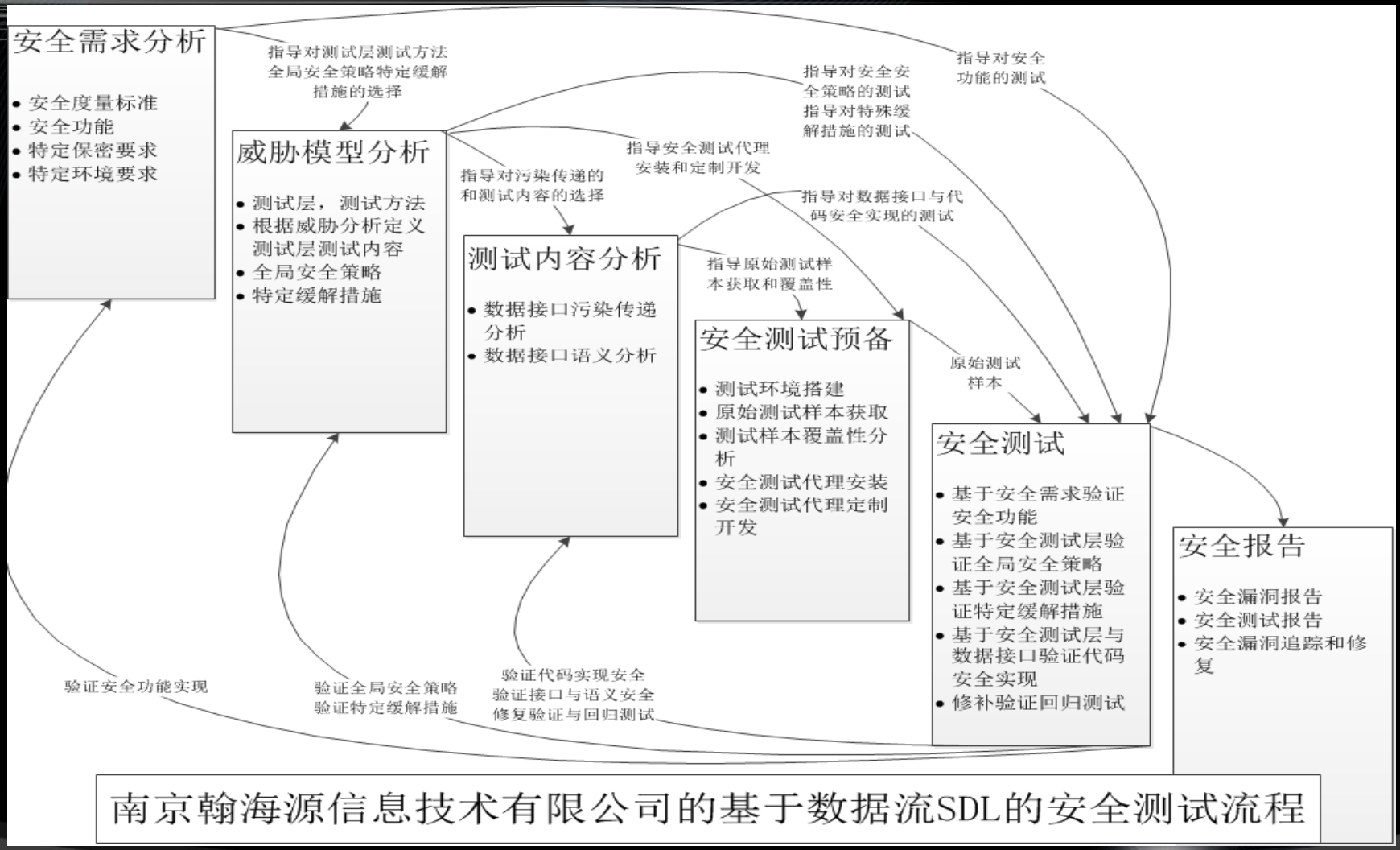
全局安全策略验证测试

- 验证
  - 定义的全局安全策略是否正确实现
  - 定义的全局安全策略是否符合默认配置
  - 定义的全局安全策略是否全局有效
  - 定义的全局安全策略是否可以被绕过
  - 定义的全局安全策略的配置数据是否有正确访问权限保护

实现安全性测试

- 验证
  - 定义的特定缓解措施是否在指定的区域正确实现
  - 定义的特定缓解措施是否在特定区域默认生效
  - 定义的特定缓解措施是否可以被绕过
  - 是否存在其他未意料到的安全问题

安全测试流程



南京翰海源信息技术有限公司的基于数据流SDL的安全测试流程

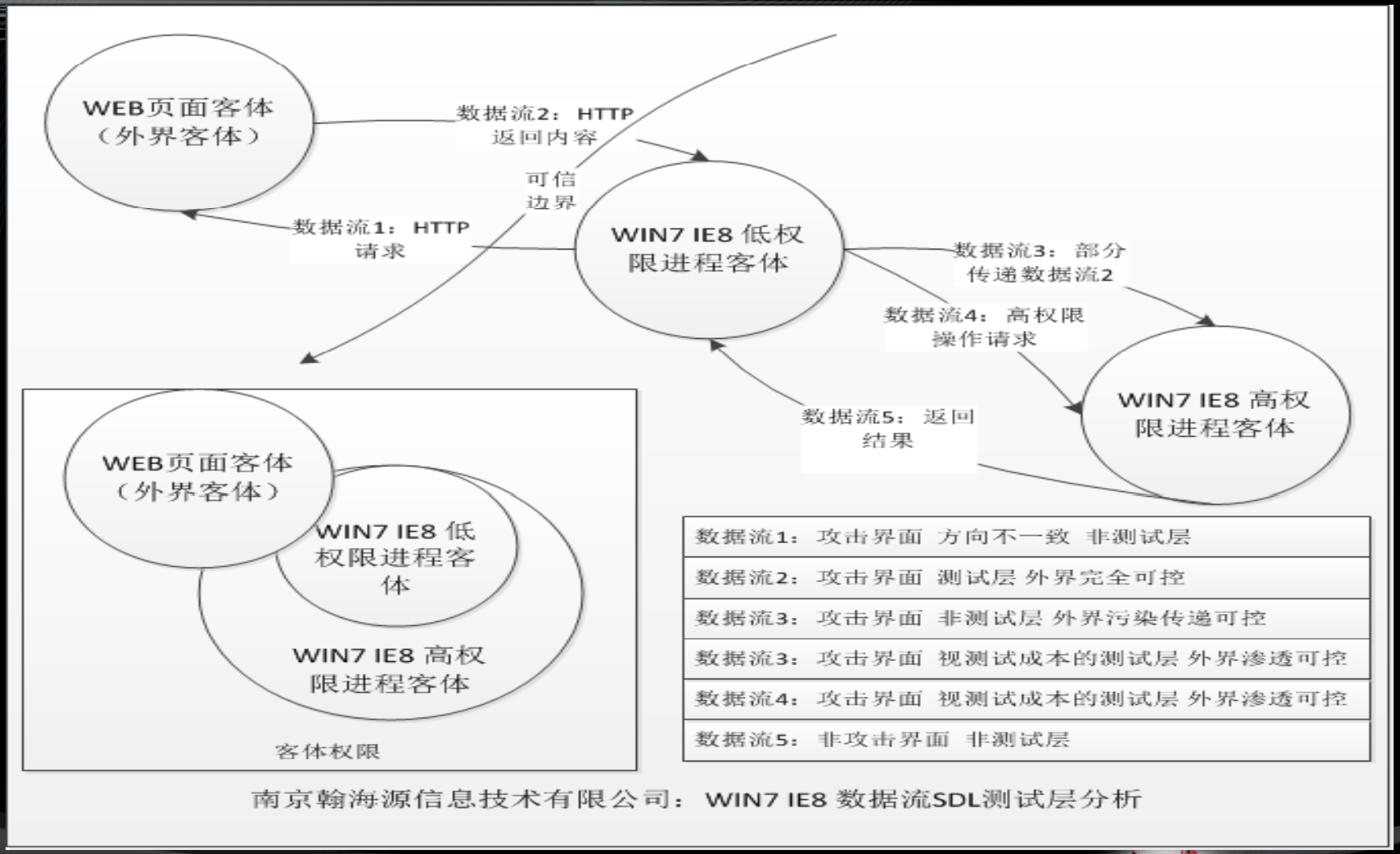
代码安全实现测试

- 基于数据流SDL分析
  - 测试区域
    - 测试层
    - 测试内容
    - 测试策略
  - 测试方法和工具（产品）选择
    - 安全度量性要求
    - 客户特定保密要求
    - 客户特定环境要求
    - 工具和产品支持

安全测试层

- 安全测试层
  - 数据流通道
    - 网络
    - 文件
    - 消息
    - IO
    - 界面操作
    - ....
  - 方向
    - 发起
    - 会话
    - 协议/状态
  - 类别
    - 外界完全可控
    - 外界渗透可控
    - 外界污染传递可控
  - 需求定义
    - 安全度量指标
    - 成本

安全测试层实例


WEB页面客体
（外界客体）
数据流2：HTTP
返回内容
可信
边界
数据流1：HTTP
请求
WIN7 IE8 低权
限进程客体
数据流3：部分
传递数据流2
数据流4：高权限
操作请求
WIN7 IE8 高权
限进程客体
数据流5：返回
结果
WEB页面客体
（外界客体）
WIN7 IE8 低
权限进程客
体
WIN7 IE8 高权
限进程客体
客体权限
数据流1：攻击界面 方向不一致 非测试层
数据流2：攻击界面 测试层 外界完全可控
数据流3：攻击界面 非测试层 外界污染传递可控
数据流3：攻击界面 视测试成本的测试层 外界渗透可控
数据流4：攻击界面 视测试成本的测试层 外界渗透可控
数据流5：非攻击界面 非测试层
南京翰海源信息技术有限公司：WIN7 IE8 数据流SDL测试层分析

安全测试内容

- 外界完全可控安全测试层
  - 所有外界可以传递的数据为安全测试内容
- 外界污染传递可控安全测试层
  - 所有外界可以污染传递的数据为安全测试内容
- 外界渗透可控安全测试层
  - 所有外界可以传递的数据为安全测试内容
  - 所有可以传递的可信边界内的内部数据为安全测试内容

安全测试策略

- 取值
  - 操作类型：参数，变量，返回值
  - 数据值类型
  - 取值范围
- 数据关联关系
  - 局部数据关系
  - 全局数据关系
  - 关系逻辑
- 逻辑语义
  - 数据逻辑含义
  - 数据时序逻辑

安全测试方法

- 黑盒
  - 针对二进制
  - 以数据边界值/有效等价类为基准
- 灰盒
  - 针对二进制
  - 以路径，条件或逻辑覆盖为基准
- 白盒
  - 针对源代码
  - 以路径，条件或逻辑覆盖为基准

安全测试方法

- 黑盒安全测试
  - 黑盒FUZZ：依赖安全经验选取边界值有效等价类，无法处理数据外部关系和逻辑语义，覆盖性依赖样本
  - 局部数据结构SMART FUZZ：部分依赖描述选取边界值有效等价类，无法处理数据外部关系和逻辑语义，覆盖性依赖样本
  - 全局数据结构安全测试：全部依赖描述选取边界值有效等价类，可处理数据外部复杂关系和逻辑语义，覆盖性不依赖样本

安全测试方法

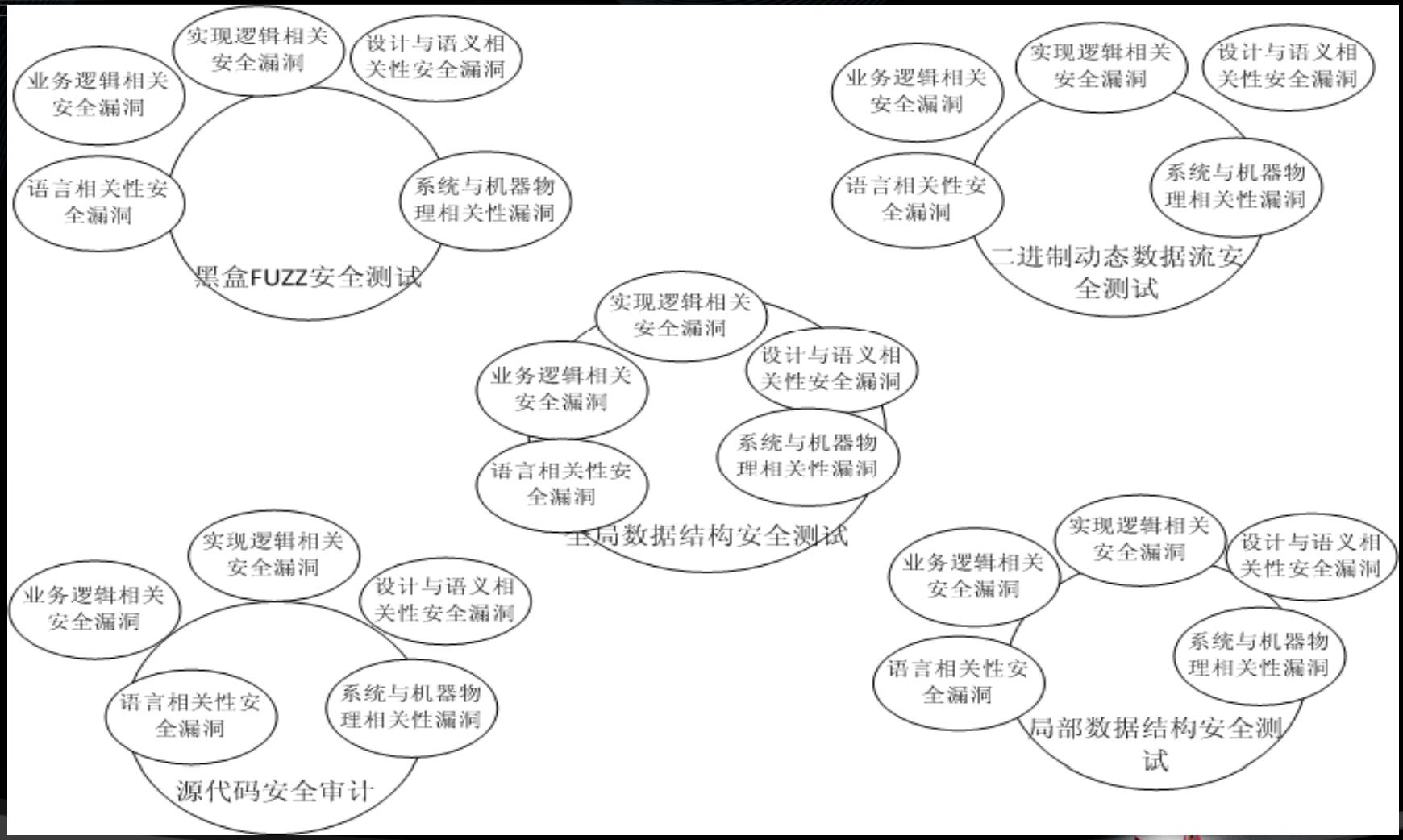
- 灰盒
  - 二进制动态数据流污染安全测试：依赖执行路径选取路径覆盖，无法处理外部复杂路径，覆盖性依赖样本
- 白盒
  - 源代码动态数据流污染安全测试：依赖执行路径选取路径覆盖，无法处理外部复杂路径，覆盖性依赖发现的可疑点

问题：当前的代码审计工具算白盒测试吗？

是语句覆盖吗？没有语句之间执行关联的流程，难以断言。分支，条件，分支条件，路径更是以动态为准则。

难以发现定性安全问题

各测试方法的覆盖领域


实现逻辑相关安全漏洞
设计与语义相关性安全漏洞
业务逻辑相关安全漏洞
语言相关性安全漏洞
系统与机器物理相关性漏洞
黑盒FUZZ安全测试
实现逻辑相关安全漏洞
设计与语义相关性安全漏洞
业务逻辑相关安全漏洞
语言相关性安全漏洞
系统与机器物理相关性漏洞
二进制动态数据流安全测试
实现逻辑相关安全漏洞
设计与语义相关性安全漏洞
业务逻辑相关安全漏洞
语言相关性安全漏洞
系统与机器物理相关性漏洞
全局数据结构安全测试
实现逻辑相关安全漏洞
设计与语义相关性安全漏洞
业务逻辑相关安全漏洞
语言相关性安全漏洞
系统与机器物理相关性漏洞
源代码安全审计
实现逻辑相关安全漏洞
设计与语义相关性安全漏洞
业务逻辑相关安全漏洞
语言相关性安全漏洞
系统与机器物理相关性漏洞
局部数据结构安全测试

安全测试覆盖性分析

- 二进制方法
- 源代码分析
- 全局数据接口方法

安全漏洞报告

- 安全漏洞分析
  - 机理与关键代码
  - 可利用性与利用场景
  - 危害性
  - POC
- 安全漏洞修复建议
  - 代码修补点
  - 特定安全缓解措施建议

安全漏洞追踪和修复

- 安全漏洞追踪管理
  - 提交接口人
  - 接口人提交开发团队
  - 开发团队确认安全漏洞（必要时指导）
  - 开发团队确认修补计划和责任人
  - 修补安全漏洞并提交安全测试

修复回归安全测试

- 回归安全测试
  - 原始安全漏洞是否被修补
  - 是否有其他绕过修补的方法
  - 修复是否带来了其他的安全问题
- 安全测试修复
  - 安全漏洞修复确认
  - 安全漏洞归档，作为下个版本优先测试的目标

安全测试安全性度量

- 覆盖性分析
- 漏洞类型分布
- 漏洞区域分析
- 漏洞一致性分析

安全测试报告

- 安全漏洞汇总分析
- 安全漏洞修复情况
- 非安全BUG汇总
- 安全测试日志统计
- 安全度量分析
  - 主要安全漏洞与原理与分布
  - 安全改进建议
    - 安全编码改进建议
    - 全局安全策略改进建议
    - 安全功能改进建议
    - 安全过程改进建议
    - 同类漏洞改进和规范定义

安全测试平台


配置管理
测试代理
自动化引擎
分析引擎
自动化安全测试框架
POC生成安全验证测试与分析
黑盒FUZZ测试策略框架
全局数据结构安全测试策略框架
二进制动态数据流污染灰盒安全测试框架
源代码动态数据流污染白盒安全测试框架
测试管理系统
覆盖性，安全性度量分析引擎
样本收集引擎
外围支撑系统
翰海源信息技术有限公司安全测试平台体系架构

总结

- 安全测试必须实施SDL中的重要环节
- 安全测试不仅仅是提供给客户安全测试
  - 改进安全
  - 度量安全
- 安全测试还要帮助客户改进开发过程
  - 推进安全意识
  - 推进开发团队安全能力
  - 逐步帮助开发团队实施SDL中的重要环节

Q/A

- 问题回答