

项目描述

需求描述

运营商骨干网上采集现网流量流向信息，根据这些原始信息检测账号是否存在异常，如果多个终端使用同一个宽带账号，超过一定阈值则触发报警机制，例如阈值为5时，同一个账号同时连接的终端数量不能超过该值，如果超过则报警。

检测方法

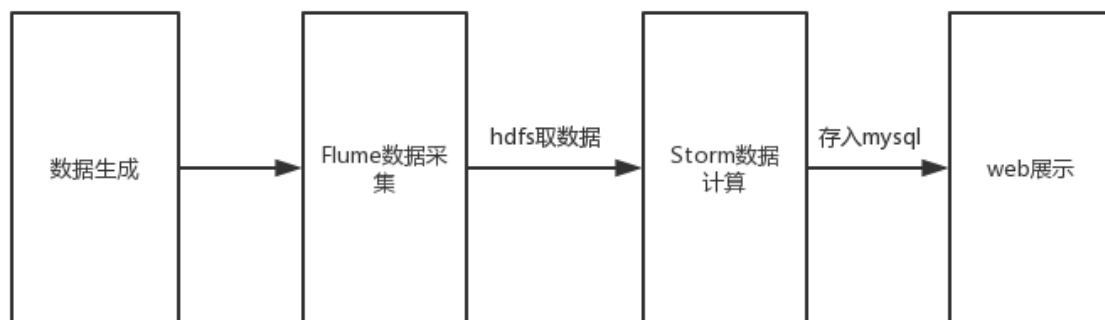
5分钟内, 同一账号,只要满足下面的任意一个条件,表示出现共享账户问题

- natIP去重求和数 > 5
- qqid去重求和书 > 20
- cookieValue + devName + osName 去重求和 > 5

| 序号 | 字段 | 类型 | 描述 |
|----|-------------|------|---------|
| 1 | time | Byte | 数据统计时间 |
| 2 | userAccount | Byte | 宽带账户 |
| 3 | userIP | Byte | 用户以太网IP |
| 4 | qqid | Byte | QQ号 |
| 5 | natIP | Byte | 内网IP |
| 6 | cookieValue | Byte | Cookie值 |
| 7 | devName | Byte | 设备名称 |
| 8 | osName | Byte | 操作系统名称 |

项目设计

项目主要分为4大模块, 分别为:

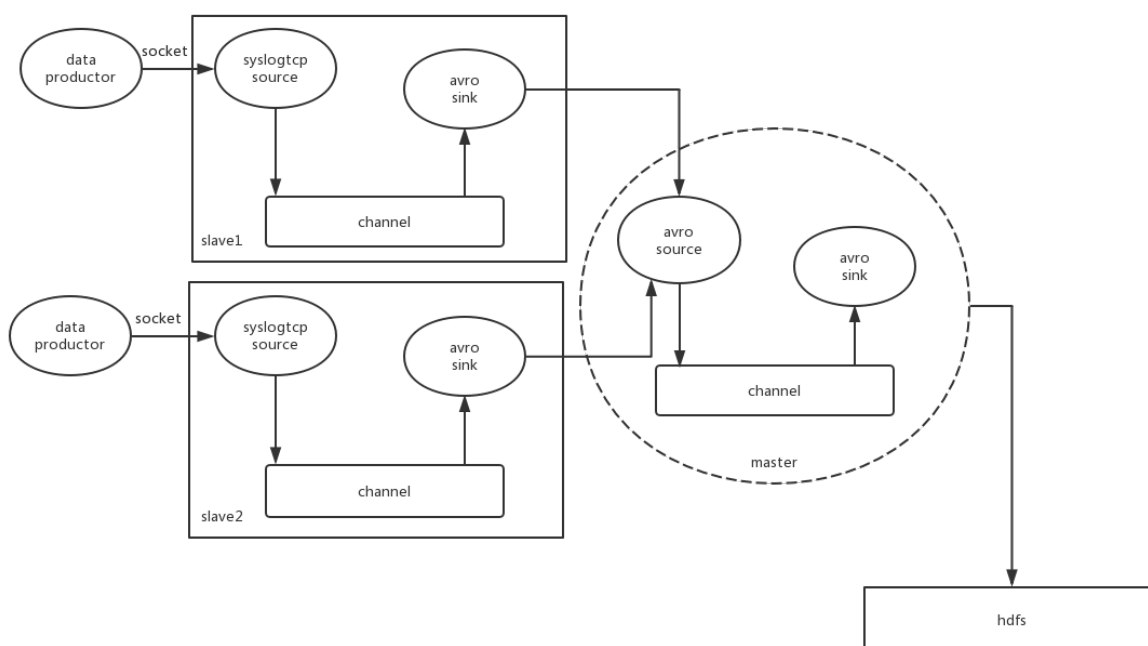


hdfs

- 分布式文件系统
- 基于流模式访问
- 数据块(block) 128M
 - 使用块抽象而不是文件可以简化存储子系统
 - 块非常适合于数据备份和
 - Master/Slave架构
 - NameNode是主节点，存储文件的元数据如文件名，文件目录结构，文件属性（生成时间,副本数,文件权限），以及每个文件的块列表以及块所在的DataNode等等
 - DataNode在本地文件系统存储文件块数据，以及块数据的校验

Flume

海量日志采集系统: agent由3部分组成:Source, Channel, Sink.

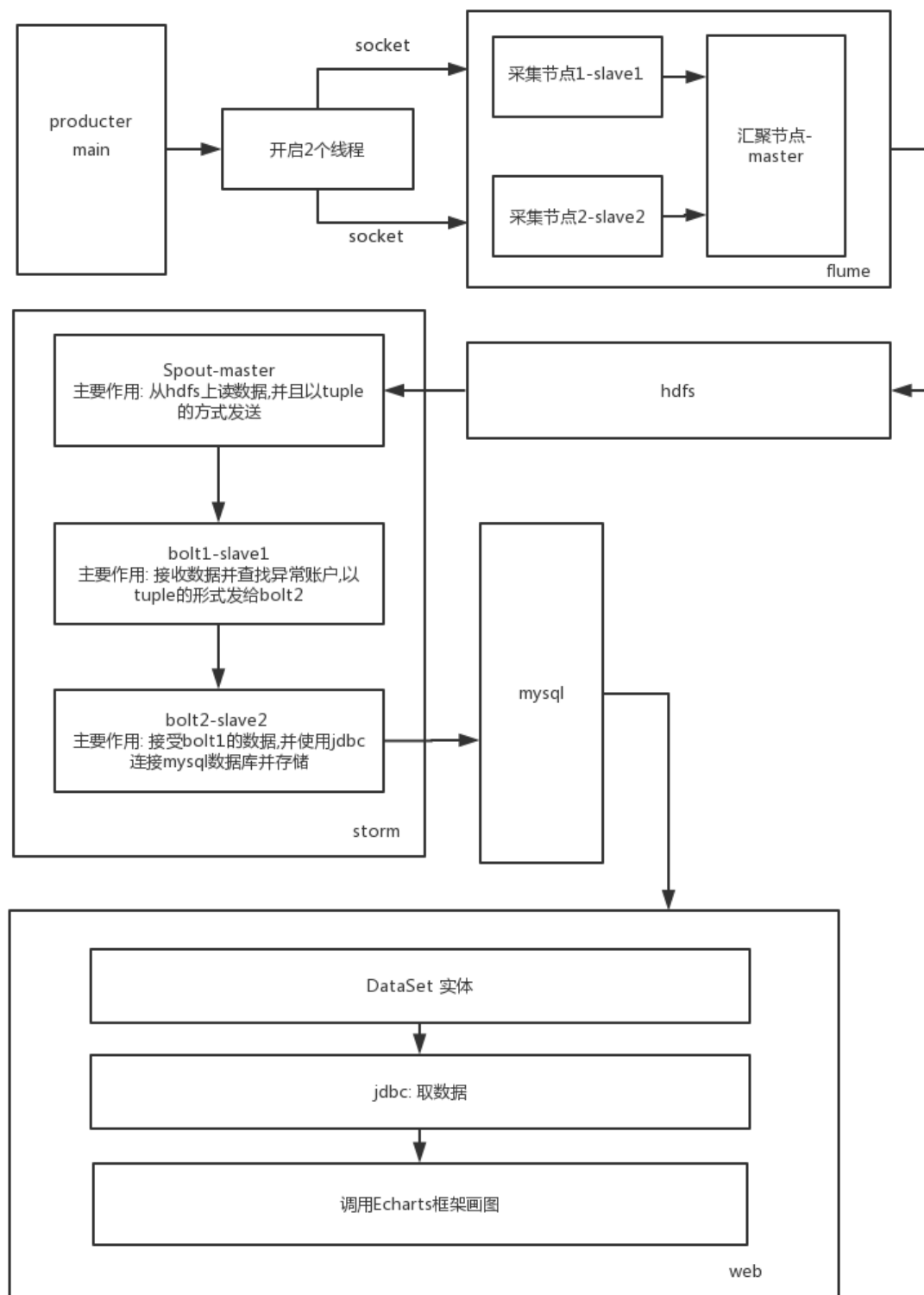


Storm

- 实时流计算框架

- nimbus (master): nimbus把分配好的任务计划信息保存到zookeeper中
- supervisor(slave...): 负责监听分配给他的工作, 根据需要启动或关闭worker进程
- Tuple
 - Spout是Storm中Topology的消息的生产者,就是tuple的创造者
Spout主动发射消息,相对Bolt来说它是一个主动的角色
 - Bolt是消息的处理者,它是一个被动的角色,因为它不能主动的发送元组, 只能接受其他Bolt或者Spout发射过来的元组, 它会对元组进行处理,然后生成新的元组继续发射给下一个Bolt, 或者它处理完元组之后,不再发射.

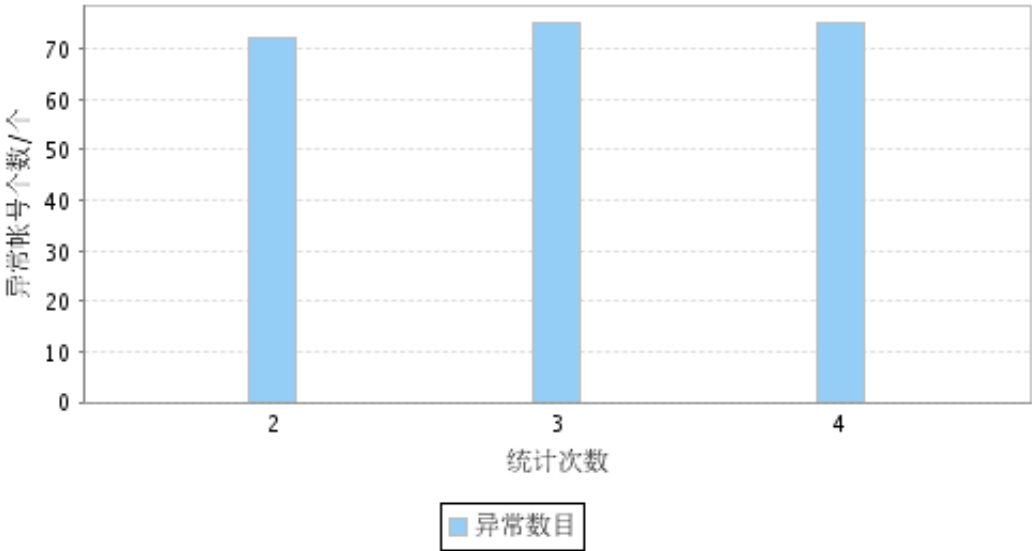
系统架构



页面展示

异常账户统计报告

异常帐号报表



异常账户统计报告

异常帐号报表

