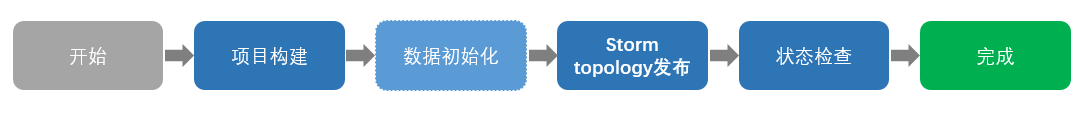
# TES部署操作手册

1. **大数据集群概述**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IP** | **域名** | **内存** | **CPU** | **存储** |
| 172.26.119.69 | catl-tes-ambari.catlbattery.com | 64G | 32C | 100G+1T |
| 172.26.119.73 | Catl-tes-kafka01.catlbattery.com | 64G | 32C | 100G+1T |
| 172.26.119.74 | Catl-tes-kafka02.catlbattery.com | 64G | 32C | 100G+1T |
| 172.26.119.75 | Catl-tes-kafka03.catlbattery.com | 64G | 32C | 100G+1T |
| 172.26.119.67 | Catl-tes-redis01.catlbattery.com | 32G | 16C | 100G+1T |
| 172.26.119.68 | Catl-tes-redis02catlbattery.com | 32G | 16C | 100G+1T |

1. **TES大数据平台发布流程**



发布流程

2.1 环境部署

环境部署仅在平台第一次上线需要。详细环境安装部署手册请参见HDP部署手册，请确保以下所需软件的版本和实际安装的软件版本一致。

**所需软件清单**

|  |  |  |
| --- | --- | --- |
| 名称 | 版本要求 | 说明 |
| HDP | 2.5.3.0 | 包含Kafka 0.10.0，Storm 1.0.1 |
| Ambari | 2.5.1.0 |  |
| Linux | Redhat6.7 |  |
| Redis | 3.2 | 编译安装 |
| Java | 1.7 |  |
| Twemproxy | 0.4.1 | 编译安装 |
| Python | 2.6 | 系统自带 |
| GCC | 4.5 | 系统自带，编译安装软件所需 |
|  |  |  |

2.1.1 软件安装相关规范

**Redis & Twemproxy 端口命名规范**

|  |  |  |
| --- | --- | --- |
| 服务名称 | 端口 | 说明 |
| Redis | 8000 | 从8000段端口开始 |
| Twemproxy | 18000 | 从18000段端口开始 |
| Kafka | 6667 |  |
| Storm | 6500-7000 | 从6500-7000段端口 |

注：其他软件安装规范详见安装手册。

**2.1.2各组件操作**

1).Redis

Redis安装在172.26.119.67和172.26.119.68两台服务器上，每台服务器3个redis实例,redis安装路径:/usr/redis;Redis配置文件地址:/etc/redis-8000.conf、/etc/redis-8001.conf、/etc/redis-8002.conf

eg、启动redis实例命令: redis-server /etc/redis-8000.conf

查看redis进程:ps –ef | grep redis

2).Twemproxy

Twemproxy同样安装在172.26.119.67和172.26.119.68两台服务器上，每台服务器一个Twemproxy代理，用作高可用，Twemproxy安装路径/usr/local/nutcracker.日志目录:

/var/log/nutcracker/server.log.

eg、启动代理:

cd /usr/local/nutcracker/sbin/

./nutcracker -d -c ../conf/nutcracker.yml -o /var/log/nutcracker/server.log

3).kafka

Topic规范:本项目8个topic，每个topic采用3个分区、3个备份。

命名规则：TES\_DATA01、TES\_DATA02、TES\_DATA03、TES\_DATA04、TES\_DATA05、TES\_DATA06、TES\_DATA07、TES\_DATA08.

Kafka相关命令:

进入 /usr/hdp/2.5.3.0-37/kafka/ 目录

cd /usr/hdp/2.5.3.0-37/kafka

1. 创建topic：

bin/kafka-topics.sh --create --zookeeper 172.26.119.73:2181 --replication-factor 3 --partitions 3 --topic TES\_TOPIC01

1. 查看所有topic

bin/kafka-topics.sh --list --zookeeper 172.26.119.73:2181,172.26.119.74:2181,172.26.119.75:2181

1. 每个topic信息描述

bin/kafka-topics.sh -describe -topic TES\_TOPIC01 --zookeeper 172.26.119.73:2181

2.2 项目构建

TES项目基于Maven构建，项目构建有两种方式，第一种是手工执行Maven命令进行项目构建，第二种是基于Jenkins进行项目的管理和构建，在本项目中推荐采用第二种。

TES Storm项目构建命令

*mvn clean package install*

Jenkins：<https://github.com/kohsuke/jenkins>

2.3 数据初始化（可选操作）

通过流程管理工具，将测试的主数据信息等相关数据刷新到Redis缓存，Storm topology将自动加载缓存数据用于实时计算。

2.4 Storm topology发布

2.4.1 构建上传

将构建好的项目Jar文件上传到Storm集群任意节点；

2.4.2 发布

执行Storm发布命令，等待拓扑发布完成后控制台会输出成功的日志信息；

1.启动命令:

storm jar <JAR\_FILENAME> <TOPOLOGY\_CLASS> -workers <WORKERS\_NUM> -calc <CALC\_NUM>

eg:

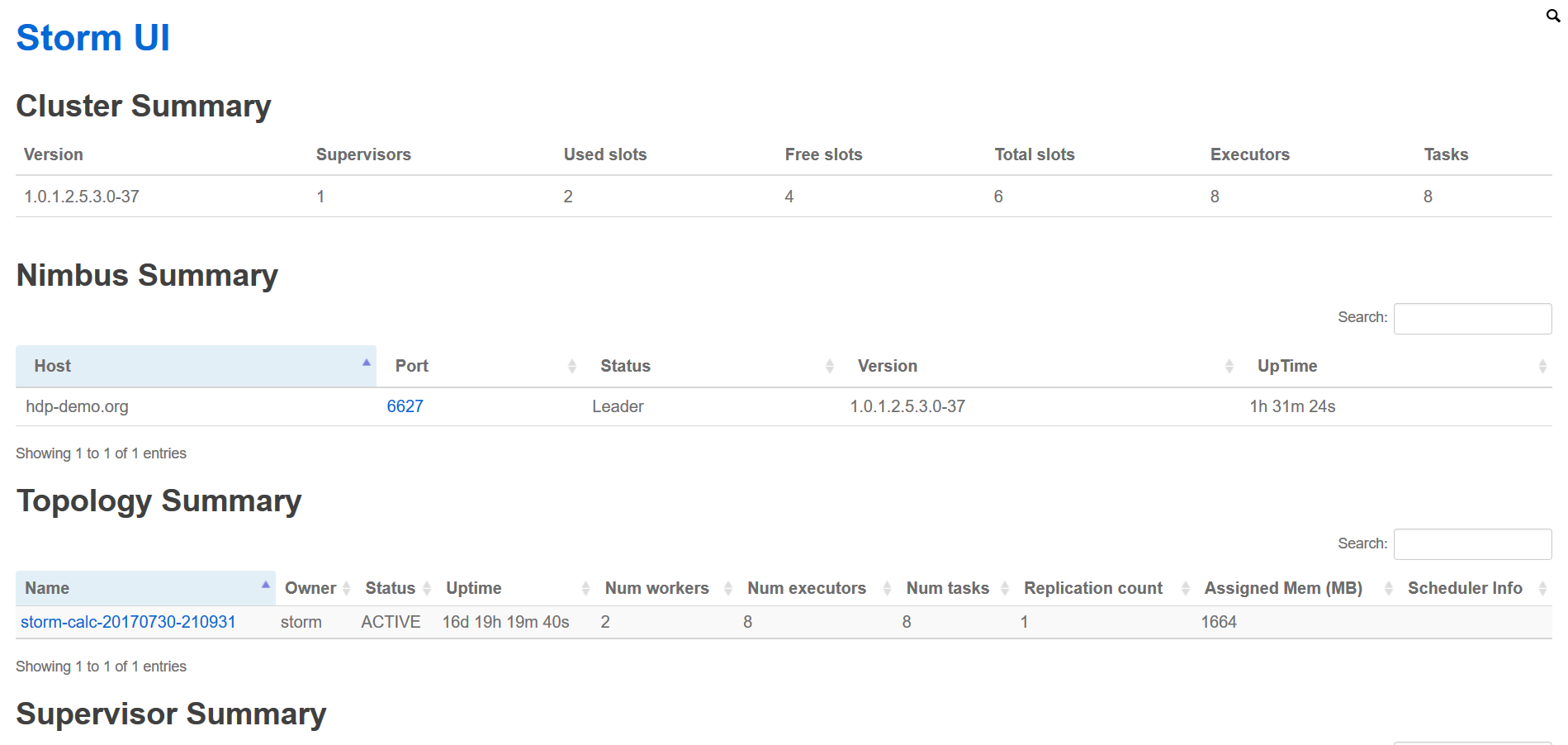
storm jar storm-service-test-28.jar com.edcs.tds.storm.topology.calc.CalcTopology -workers 45 -calc 100

**Storm命令行参数说明**

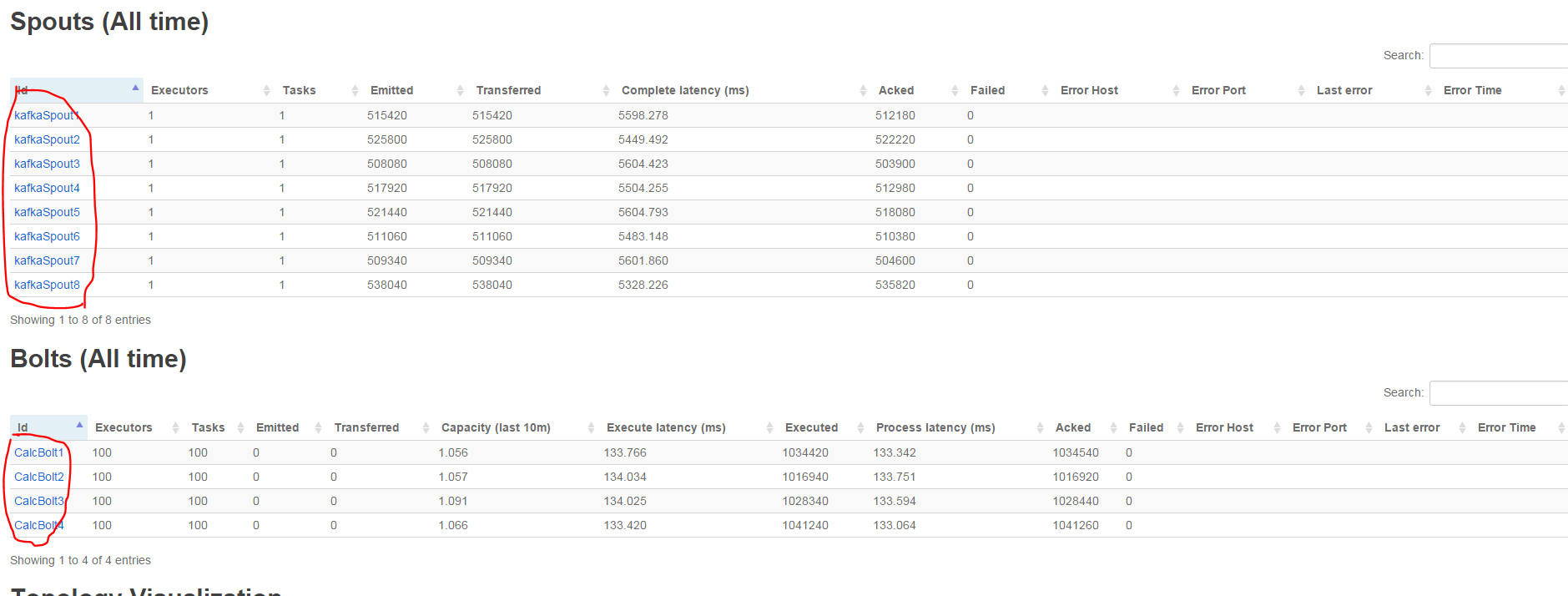
|  |  |  |
| --- | --- | --- |
| 命令行参数 | 描述 | 建议 |
| JAR\_FILENAME | Topology Jar文件路径 |  |
| TOPOLOGY\_CLASS | Topology 类路径 |  |
| WORKERS | Wokers数量 | 默认值1，建议根据并发情况设置 |
| CALC\_NUM | Calc并发计算线程数量 | 默认值1，建议根据并发情况设置 |

2.5 状态检查

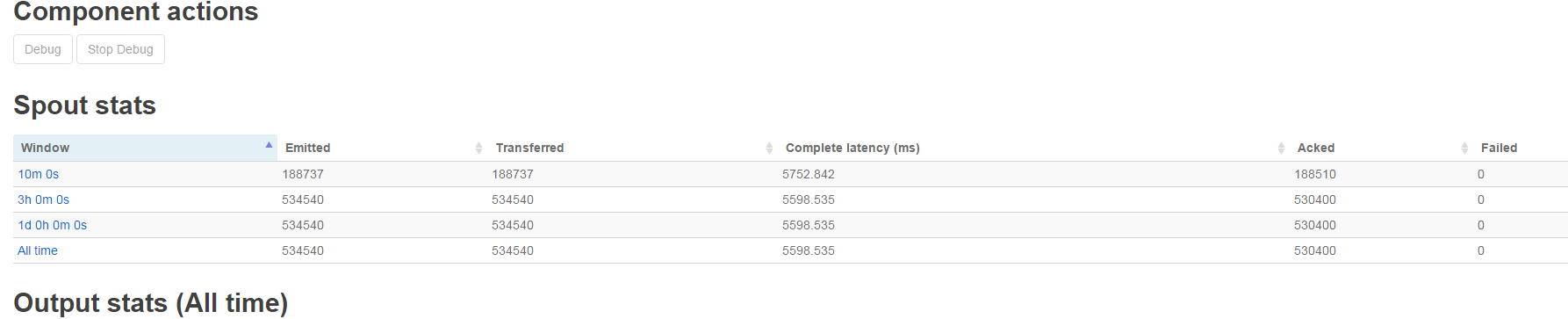
**打开Storm UI**，刷新页面后会看到已经发布成功的拓扑，点击拓扑名称进入拓扑主页，在拓扑主页包含了拓扑相关的状态信息，节点状态以及配置信息，如果拓扑发布成功，我们可以看到拓扑的状态为Active，并且已经开始接收数据进行计算。关于Storm集群服务的详细日志，可以通过Log Search进行浏览或搜索。

Storm UI

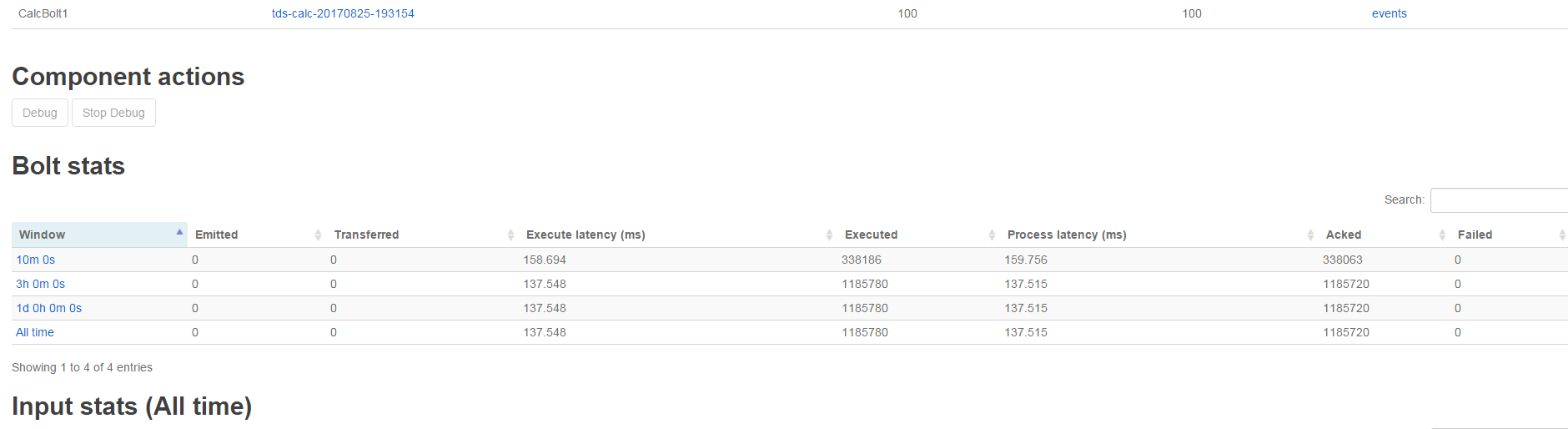
**点击Topology Summary**下面的Name,进入查看Topology详情

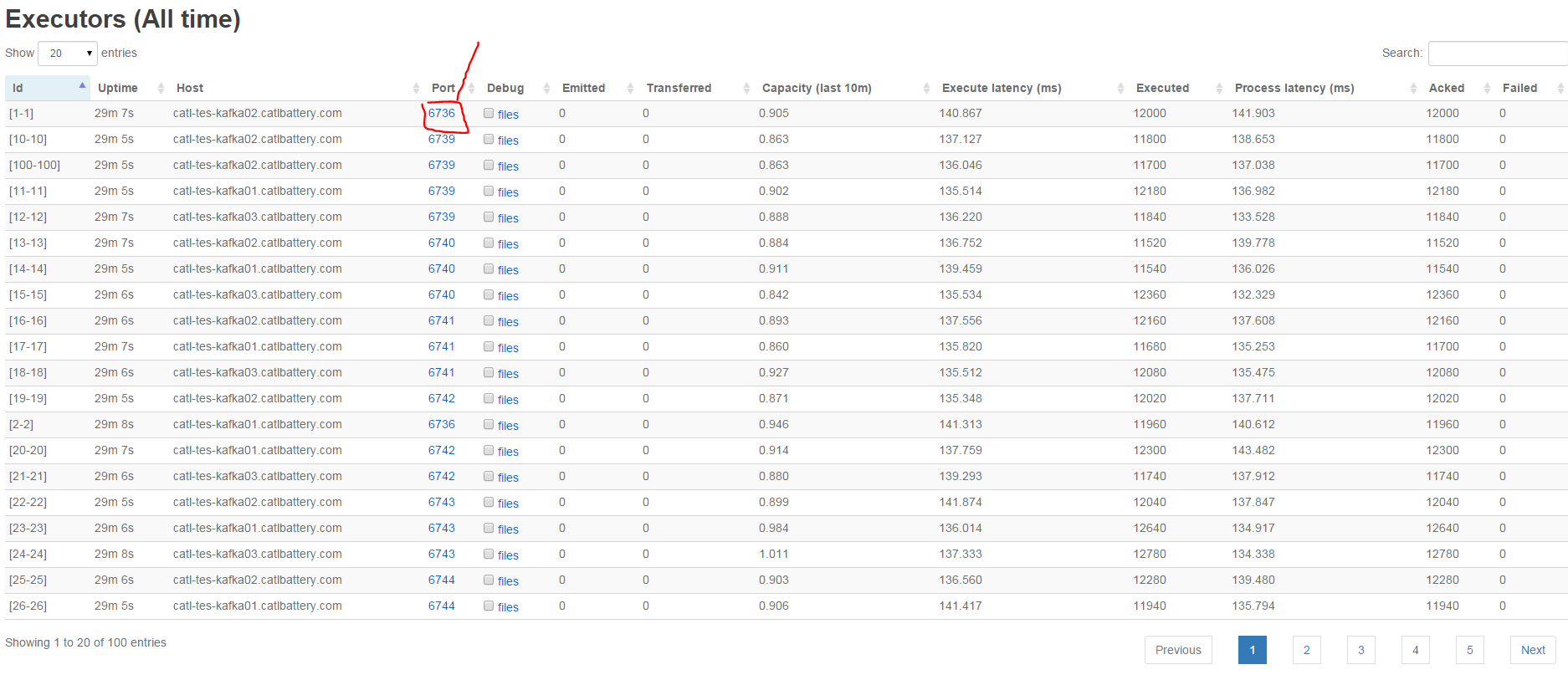


A.分别点击spout，可以查看最近一段时间，spout处理的数据量



B.分别点击bolt,可以查看最近一段时间bolt处理的数据量，以及每个bolt对应的进程处理的数据量和每个进程的日志。

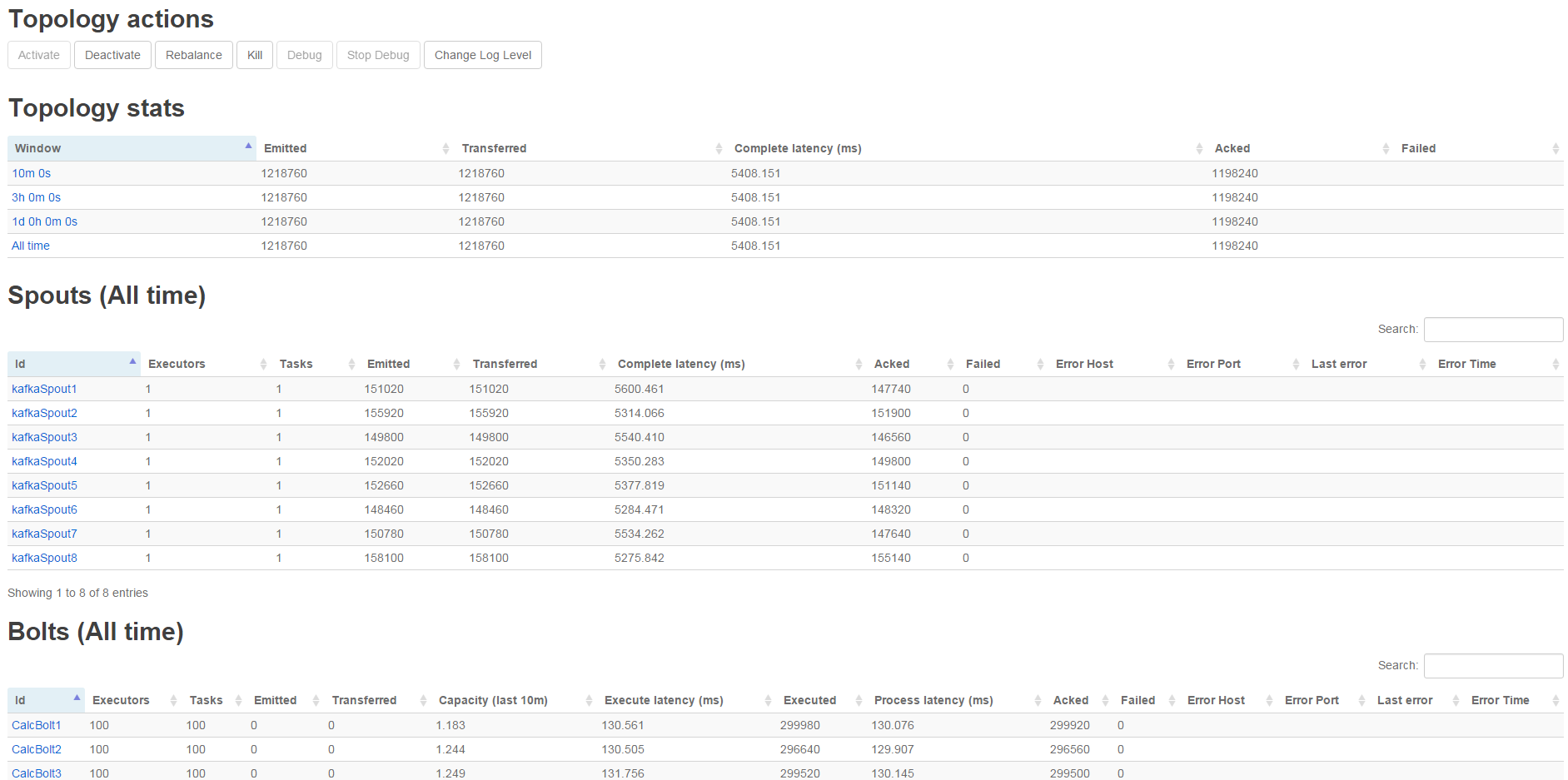




1. **异常排查流程**

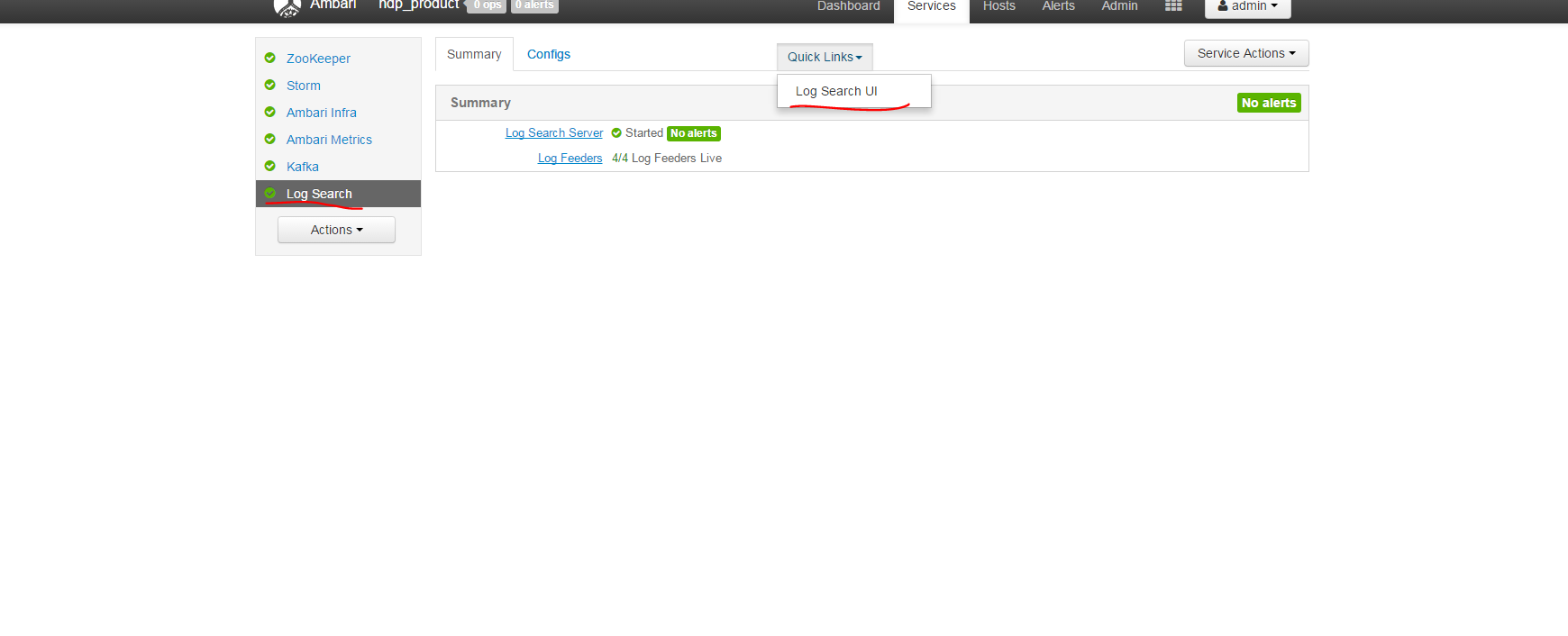
3.1 检查拓扑状态

打开Storm UI，检查拓扑的状态，观察拓扑处理消息的频率（TPS）以及各个节点的状态；



3.2 检查服务状态

登录Log Search，检查Storm、Kafka服务的日志，排查是否存在异常日志信息；



3.3 常见异常

3.2.1 拓扑计算的延迟率很高（处理单条消息耗时超过15秒以上）

造成这个问题的原因有三种：

1、网络中断或高延迟，排查网络因素，网络恢复正常后，造成的影响会逐步消退；

2、Storm rebalance，通过rebalance可以让topology重新分配资源，平衡负载压力，或者进行topology的自动故障转移，这个功能是Storm的高可用特性，通过人工操作或服务自身自动触发。这个过程中，topology会经历下线和上线两个环节，此时Kafka的消息会积压，等待topology恢复正常后会继续进行消息处理计算，造成的影响会逐步消退；

3、集群服务重启阶段，这个过程中，Storm集群服务会整体重启，等待topology恢复正常后会继续进行消息处理计算，造成的影响会逐步消退；

3.2.2 Storm UI界面显示拓扑发生异常

建议根据异常信息判断故障原因并且排除异常，异常处理完成后重新发布topology即可。

注：在TES Storm中会将无法处理、致命的异常抛出到顶层（RuntimeExceptions），一般这类异常包括配置数据格式异常、初始化失败、IO异常等致命错误，这类错误发生在集群环境第一次初始化阶段、Storm topology发布阶段，因为人为错误或环境等因素导致，需要人工干预排除异常。

1. **附录**

4.1 Storm官方文档

http://storm.apache.org/releases/1.0.3/index.html

4.2 Kafka官方文档

http://kafka.apache.org/0100/documentation.html

4.3 Ambari官方文档

https://hortonworks.com/apache/ambari/