大数据自主部署简要设计

# 背景

当前大数据主要使用的是Cloudera Manager来部署管理Hadoop集群，但是CM存在以下几个问题：

* 只支持预定的Hadoop版本
* 对多集群交叉部署支持不友好
* 不能增加额外的组件，比如Livy
* 升级组件代码麻烦
* 大集群下性能不好

我们构建了基于SaltStack的部署系统，以命令行方式执行部署，但是系统也存在其他问题：

* 没有UI系统，使用不方便
* 缺少权限控制
* 没有审计，无法跟踪查询
* 等等

# 解决方案

我们需要构建一套系统来替换CM：

* 我们已经有一套基于Prometheus的监控系统来替代CM的指标监控功能
* 当前我们要开发一套部署系统来替代CM的部署功能
* 将来我们可能还需要一套系统来替代CM的服务状态监控功能

本文主要聚焦于快速构建一个可用的部署系统，并将来在此基础上进一步完善。

# 架构

集群信息

同步配置

审计日志/配置组

保存配置

判断权限

发送命令

拉取脚本

Salt Minion

Salt Minion

Salt Minion

文件服务器

GitLab

集群元数据

MySQL

自主部署Web

Salt Master

权限

分发命令

拉取配置

### 用户案例

增加节点：

1. 在集群元数据系统里添加节点和节点上的组件
2. 在自主部署系统里找到节点，对组件执行安装命令
3. 添加组件安装日志

修改配置：

1. 在自主部署里打开对应集群信息，修改保存配置文件
2. 配置文件保存到GitLab
3. 添加配置文件修改日志

执行命令：

1. 自主部署上触发命令执行
2. 自主部署从元数据系统获取要执行的机器列表
3. 将机器列表和要执行的命令发送给Salt Master
4. Salt Master从GitLab获取命令详情，转发给机器的Salt Minion
5. Salt Minion从重建服务器获取配置，并执行命令

控制权限：

1. 在神盾上，根据 组件->角色（NameNode，DataNode）->集群 的维度进行授权

# 实现设计

### 数据库

config\_group

* id bigint
* cluster\_type varchar
* cluster\_name varchar
* role\_type varchar
* name varchar //cluster\_type,cluster\_name,role\_type,name四个字段一起唯一
* created\_by varchar
* created\_date timestamp
* last\_modified\_by varchar
* last\_modified\_date timestamp

config\_group\_host

* id bigint
* config\_group\_id bigint
* host\_name varchar
* last\_configs\_md5 //记录最后一次更新这个节点时，复制过去的配置文件的MD5
* created\_by varchar
* created\_date timestamp
* last\_modified\_by varchar
* last\_modified\_date timestamp

operation\_log

* id bigint
* cluster\_type varchar
* cluster\_name varchar
* role\_type varchar
* operation\_user varchar
* operation\_type varchar
* operation\_target varchar
* operation\_time timestamp
* operation\_details varchar

### 基础Service

ComponentMetaService：提供组件元数据配置相关接口，比如有哪些组件类型，NameNode有哪些链接地址，支持哪些操作等

ConfigGroupService: 提供配置组相关服务方法

OperationLogService: 提供操作日志相关服务方法

ClusterService：对接chaos集群管理，封装获取集群信息的相关接口

GitService：提供对配置git配置操作相关接口

SaltStackService：封装SaltStack操作，提供对集群操作的相关接口

### 前端接口

ComponentMetaController：对应封装ComponentMetaService对前端的接口

ConfigGroupController：对应封装ConfigGroupService对前端的接口，提供配置组查询，修改接口

ClusterController：封装ClusterServic，SaltStackService对前端的接口，提供集群列表，以及集群节点列表查询，提供节点状态信息，集群操作，节点操作接口。对应集群-节点页面。

ConfigController：封装GitService对配置的相关操作。对应集群-配置页面。

OperationLogController：封装OperationLogService对日志相关操作。对应集群-日志页面。

# 组件元数据

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 组件 | 角色 | 集群操作 | 主机操作 | 备注 |
| HDFS | NameNode |  |  |  |
| DataNode |  |  |  |
| JournalNode |  |  |  |
| ZKFC |  |  |  |
| YARN | ResourceManager |  |  |  |
| NodeManager |  |  |  |
| Hive | HiveServer2 |  |  |  |
| HiveMetastore |  |  |  |
| UDF | UDF |  |  | 需要一个发布到HDFS的操作 |
| Zookeeper | Server |  |  |  |
| Livy | LivyServer |  |  |  |
| Client | Client |  |  |  |

注：Client第一期可以先不支持