

CloudConf 执行引擎需求文档

序号	版本	说明	修订人	修订日期	备注
1	v0.1	创建文档	王苏	2018-6-27	整理需求

CloudConf 执行引擎需求文档

1. 产品背景
2. 可行性分析
3. 定义功能描述
 - 3.1 架构图
 - 3.2 功能模块
 - 3.3 功能需求
4. 非功能需求
 - 4.1 部署
 - 4.2 稳定性
 - 4.3 安全性
 - 4.4 性能
5. 约束条件
 - 5.1 有无代理功能对比
 - 5.2 支持操作系统列表

1. 产品背景

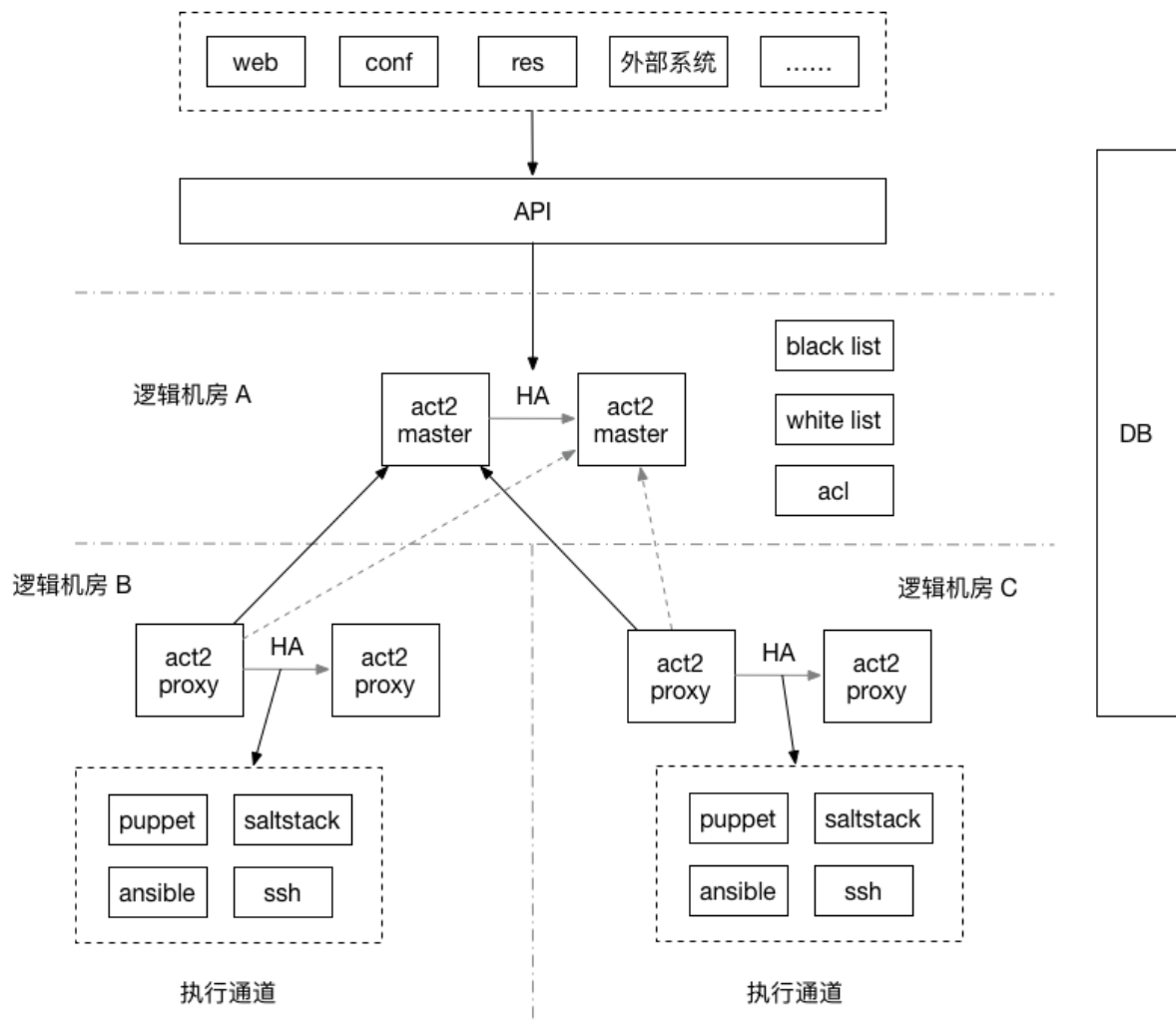
执行引擎作为自动化产品的核心能力是必不可少的。目前 CloudConf 依赖 ST2 作为自己的执行引擎，在落地过程中遇到了很多问题，包括对稳定性和性能的挑战。同时，针对企业级用户提出了更高的要求，包括命令黑白名单和访问控制等，显然开源的 ST2 已经满足不了需求，需要我们自己设计并实现一套执行引擎

2. 可行性分析

执行引擎是自动化产品的核心，既可以作为组件被内部产品调用，又可以单独作为产品暴露给外部客户，提供自动化操作的能力。目前外部客户对底层执行能力的诉求比较强烈，如太平洋保险、宁波银行和新华三等，希望我们把自动化能力通过服务化的方式提供出来给上层调用，他们自己封装前台页面。执行引擎本身只提供自动化调度和执行能力，是与业务解耦的。底层的执行通道可以对接外部系统如 puppet、saltstack、ansible、ssh 等

3. 定义功能描述

3.1 架构图



3.2 功能模块

- act2-master: 负责作业的调度分发, 包括执行命令的黑白名单和用户访问控制 ACL
- act2-proxy: 负责接收 master 下发的指令, 然后转发给执行通道执行。将执行通道执行结果设备采集的信息转发给 master, 由 master 保存到数据库中

3.3 功能需求

- 作业执行: 支持作业执行, act2-master 提供 API 给外部系统调用, master 本身不包含操作定义, 需要外部系统提供待执行的设备列表、操作定义、执行用户名、密码(可选)、超时时间等
- 队列控制: 支持队列控制, 查询任务队列的状态和监控。未下发到执行通道执行的作业(在队列中排队等待), 可以被取消。已经下发到执行通道执行的作业无法取消。支持并发数量的限制, 支持执行命令的优先级控制
- 实时输出: 支持作业执行过程中实时输出, 用户可以在前台实时查看执行过程中作业输出的情况
- 异常控制: 支持异常检测、超时、重试等机制
- 查询审计: 支持查询每一个作业执行情况的审计, 方便审计人员核查

4. 非功能需求

4.1 部署

- 支持集群部署模式（HA高可用），支持水平扩展，满足负载均衡的需求

4.2 稳定性

- 支持 7*24 小时的不间断运行，在故障情况下可以自动切换和重连
- 支持 act2 master/proxy 之间的心跳检测机制

4.3 安全性

- 提供黑白名单控制，对操作定义中高危命令提供过滤
- 提供对访问者的权限控制 ACL，支持 API 级别的权限控制

4.4 性能

- 单台 act2 master/proxy 节点提供服务，满足支持 3000 台客户端并发调度

5. 约束条件

5.1 有无代理功能对比

功能	无代理 (ssh)	有代理 (puppet)	有代理 (saltstack)
文件下发	√	√	√
命令执行	√	√	√
脚本执行	√	√	√
服务命令	√	√	√
软件安装	√	√	√
设备发现		√	√
账号收集		√	√
软件包收集		√	√
服务收集		√	√
定时任务收集		√	√
补丁管理			√
实时输出			√
自定义采集			√
社区模块支持			√

5.2 支持操作系统列表

操作系统	版本	备注
Windows	2016	全部支持
	2012	全部支持
	2008	全部支持
	2003	部分支持
CentOS RedHat	7	全部支持
	6	全部支持
	5	全部支持
	4	部分支持
Suse	15	全部支持
	12	全部支持
	11	全部支持
	10	全部支持
Ubuntu	18.04	全部支持
	16.04	全部支持
	14.04	全部支持
	12.04	全部支持
AIX	7	全部支持
	6	全部支持
	5	部分支持