R1.1

目录

1 简介 1

1.1 目的 1

1.2 文档范围 1

1.3 参考 1

2 系统概述 2

2.1 概念 2

3 系统架构 3

4 界面原型 4

4.1 日志管理 4

4.2 主机情况管理 4

5 系统代码 5

5.1 运维(内部代码待定) 5

6 系统接口 6

6.1 Purus 6

7 系统模型 7

8 修改记录 8

# 简介

## 目的

本文描述了运维管理平台的概要设计。

## 文档范围

本文档阅读范围

1. 运维管理平台开发人员

## 参考

# 系统概述

## 概念

# 系统架构

运维平台

mongodb

主机

SSH(paramiko)

主机

主机

git.quxun.io

OAuth

1. 运维平台通过SSH(paramiko库)链接到目标主机，执行运维命令
2. 通过git.quxun.io使用OAuth做用户认证
3. 通过mongodb作为持久存储，保存用户信息，权限，操作日志等

# 界面原型

待补充

## 主机概况管理

## 日志管理

# 系统代码

## 运维(内部代码待定)

### core/ssh\_pool.py

#### SSHPool

主机连接池

1. get\_client(host\_id)

根据host\_id，从连接池获取（或者创建）一个可用的ssh连接

1. release\_client(client)

释放连接到连接池

#### SSHClient

1. connect
2. put\_file
3. exec\_command

### handler/api/overview.py

#### ApiOverviewHandler

1. def get(self, path)

根据path调用相应的get\_\*方法

1. def get\_overview(host\_id)

检查权限

根据host\_id从ssh池取得ssh链接

调用命令获取主机概况信息

磁盘 df 内存 free 负载 ?

解析输出，生成json，返回

1. def get\_list()

检查权限

根据用户，获取用户可操作的主机列表

# 系统接口

## Purus

### 主机列表 /api/overview/list

1. 方法：GET
2. 输入参数：

无

1. 输出参数：

{ “status”: “ok”, “data”: [

{“id”: 1, “name”: “anna”, “notes”: “一起充界面”, “ip”: “127.0.0.1”},

{“id”: 2, “name”: “better”, “notes”: “话费核心”, “ip”: “127.0.0.2”}

]}

### 获取主机概览 /api/upstream/*{host\_id}*

1. 方法：GET
2. 输入参数：

无

1. 输出参数：

{

"status": "ok",

"data": {

“disk”: [

{“disk”: “/dev/xvda1”, “use”: “74%”, “mount”: “/”},

{“disk”: “/dev/xvdb1”, “use”: “9%”, “mount”: “/data”}

],

“memory”: {

“total”: 2048472,

“used”: 1942864,

“fee”: 105608,

},

“load”: [0.15, 0.10, 0.14]

}

}

# 系统模型

## 用户user

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 说明 |
| \_id | int | 用户id |
|  |  |  |
|  |  |  |

## 主机host

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 说明 |
| \_id | int | 用户id |
| name |  |  |
| ip |  |  |
| port |  |  |

# 修改记录

|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | **版本** | **作者** | **说明** |
| 2016/2/16 | Rev 1 | 羊笑佺 | 初始版本 |