

# 接口测试网址

<http://172.20.155.150:8888/swagger-ui.html>

- 右上角可切换服务

Select a spec 授权服务

- 主要使用的服务为：

- 1.授权服务-----获取token
- 2.用户服务-----用户和Team功能接口
- 3.对接prometheus服务----指标集和指标接口

## 获取认证token

1. 通过授权服务获取token 2.具体参数如下：

POST /auth/token 获取认证token

账号:account,密码:password,type:0内部账号登录1酷开域账号登录

Parameters

Name	Description
AIOPS-TOKEN string (header)	header中AIOPS-TOKEN字段用于使用API的验证 AIOPS-TOKEN - header中AIOPS-TOKEN
loginUserVo *required (body)	loginUserVo Example ValueModel { "userName": "wudongyang@coocaa.com", "password": "wdy.0717" }
type *required integer(\$int32) (query)	type 1

Cancel

Parameter content type  
application/json

- 3.取出返回头key为aiops-token里的value在调用其他接口带上即可：

```
{
  "page": 0,
  "count": 5,
  "conditionConnection": "or",
  "orderBy": "update_time",
  "sortType": "desc",
  "conditions": [
    {
      "query": "update_time",
      "connection": "小于",
      "queryString": "2019-08-06 09:44:04"
    },
    {
      "query": "admin_user_id",
      "connection": "=",
      "queryString": "1"
    }
  ]
}
```

- 参数解释  
page:从0开始取  
count:第page页的count条记录  
conditionConnection: 可取and或or, 为下列conditions的并列条件  
orderBy: 指定返回的数据以数据库哪一列排序, 具体可选值可查看接口返回值  
sortType: 可取desc(降序)或asc(升序), 代表按orderBy的字段排序  
conditions:为list, query为数据库表列名, connection可取=、like、大于、小于, queryString为具体值, 如id=1
- 注意 如不需要复杂的模糊查询功能, 只传page和count即可
- 重要响应结果解释  
userList: 为Team下的用户, 返回第一页的前5条用户, 如需获取更多, 可以调用【分页获取某个Team下的用户】
- code: 为总记录数

## 批量删除统一规范格式如下

- 请求参数

```
{
  "items": [
    {
      "query": "id",
      "queryString": "15"
    },
    {
      "query": "id",
      "queryString": "16"
    }
  ]
}
```

- 参数解释  
items: 为list  
query为数据库表列名  
queryString为具体值, 如id=1  
建议使用唯一确定数据的条件和值

## 统一格式新建或修改接口

- 传id为修改
- 不传id或传0为新建
- 需要修改的值可传
- 不需要修改的值可不传

## 以下为针对不同服务及接口的使用说明

### 用户服务

## /team/create

- 请求参数示例

```
{
  "id": 17,
  "name": "测试Team4",
  "adminUserId": 1,
  "userIdList": "21,22"
}
```

- 参数解释

1. userIdList为新建时同时指定该Team下的成员，userId以小写逗号隔开

## /users/create

- 参数解释 用于修改某用户所属的Team { "id": 41, "teamIds": "12,17" }

- 注意

teamIds以小写逗号隔开,且id对于的Team必须在数据库存在，teamIds更新时为单纯覆盖，不会做额外处理

# 以下为新建指标及进行异常检测的调用流程

## 对接Promtheus服务下的接口说明

### 第一步：分页获取指标集选择监控的指标集

#### /kpiListing

- 响应结果解释：

```
{
  "msg": "success",
  "code": 17,
  "success": true,
  "data": [
    {
      "id": 17,
      "name": "内存使用率",
      "promExpression": "((node_memory_MemTotal_bytes%s - node_memory_Buffers_bytes%s - node_memory_Cached_bytes%s)",
      "updateTime": "2019-08-08 10:16:17",
      "createTime": "2019-08-08 10:16:17"
    },
    {
      "id": 16,
      "name": "内存Cache容量",

```

重要参数为promExpression，后面新建定时任务会用到

这里我们选取irate(node\_network\_receive\_bytes\_total[s5m])\*8进行测试

### 第二步、根据指标集promExpression获取其指定时间段内可取的条件及其可取值

#### /metrics/condition/{minute}

- metricsName : irate(node\_network\_receive\_bytes\_total[s5m])\*8
- minute : 1

Name	Description
AIOPS-TOKEN string (header)	header中AIOPS-TOKEN字段用于使用API的验证 <input type="text" value="eyJ0eXAiOiJKc29uV2ViVG9rZW4iLCJ"/>
metricsName * required string (query)	metricsName <input type="text" value="irate(node_network_receive_bytes_to"/>
minute * required integer(\$int32) (path)	minute <input type="text" value="1"/>

Execute

- 注意  
需去除掉步骤一获得的promExpression中即irate(node\_network\_receive\_bytes\_total%s[5m])\*8的所有%s才可获得正确结果
- 响应结果解释：

```

{
  "data": {
    "instance": [
      "172.16.33.2:9100",
      "172.16.33.20:9100",
      "172.16.32.219:9100",
      "172.16.33.31:9100",
      "172.16.32.181:9100"
    ],
    "baseLabels": [
      "type system module job hostname __name__"
    ],
    "device": [
      "lo",
      "eth0"
    ]
  }
}

```

baseLabels为基础标签值，一般可以忽略

此返回结果代表接下来的新建指标监控时instance条件的可取值及device的可取值

不同指标其可选条件不同

## 第三步、新建指标监控定时任务

/metrics/create/{type}

- 请求参数示例

```

{
  "id": 0,
  "metricName": "网络进带宽",
  "teamIds": "1",
  "metricsId": 13,

```

```
"taskName": "网络进带宽-实例172.16.33.2:9100",
"taskDescription": "按taskCron表达式定期获取网络进带宽并按指定模型进行异常检测",
"queryRange": {
  "query": "irate(node_network_receive_bytes_total%s[5m])*8",
  "span": 8640,
  "step": 60,
  "conditions": {
    "instance": "172.16.33.2:9100"
  }
},
"taskCron": "1 0/2 * * * ?",
"modelName": "1565079045604"
}
```

- 参数说明

**请求体：**

- id：传为修改，不传或传0为新增
- metricName：指标集名，可取自指标集列表接口的name
- teamIds：指标所属Team，可取自Team列表接口的id
- taskName：指标名，即此定时任务的名字
- taskDescription：定时任务描述，自定义
- queryRange：为对象，其参数用于异常记录是存储的数据时间范围长度
- query：取自指标集列表接口prom\_expression，不可去除%s
- span：代表异常存储时记录距检测点span时间段的数据，单位为s
- step：代表异常存储时记录每step的数据，单位为s
- conditions：取自第二步获取的条件及其取值，用于异常检测时按条件拉取数据进行检测
- taskCron：cron表达式
- modelName：取自模型列表接口的name值，具体接口联系陈

**url参数：**

- type: 0不启动定时任务1启动

至此，一个异常监控项的新建完成

## 指标异常列表展示接口说明

/metisexception

- 请求参数示例

```
{
  "page": 0,
  "count": 5,
  "conditionConnection": "and",
  "conditions": [
    {
      "query": "task_id",
      "connection": "=",
      "queryString": "39"
    }
  ]
}
```

- 参数解释  
task\_id: 取值定时任务列表中及上述步骤第三步建立的指标监控项的id  
上述请求代表获得指定指标下的异常列表

上述请求代表获得指定指标下的异常列表

- 响应结果

[illegible]

- 响应结果解释
  - recentUserId: 最近修改用户, 为-1代表不存在
  - matrixDataJson: 为异常时刻检测结果及当时的数据
  - userToReasonJson: 用户名及其修改理由JSON字符串
  - status: 0未处理1已修正2已恢复3已修复

matrixDataJson: 为异常时刻检测结果及当时的数据

status: 0未处理1已修正2已恢复3已修复

## 模型训练接口