**Linux用户账号和密码管理接口文档**

## 1 Host管理

### 1.1 新增Host

##### 接口地址

POST

<http://localhost:8080/simple/linux/add/host>

##### 请求参数：

①请求的头行：

Content-Type: application/json

②请求的Body部分(json)：

[

{"ip":"192.168.0.211","user":"root","passwd":"1","cluster":"cluster2"}

]

请求参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 必选 | 类型 | 默认值 |
| ip | 主机ip | 是 | String |  |
| user | 用户名 | 是 | String |  |
| passwd | 密码 | 是 | String |  |
| cluster | 已存在的Resource | 是 | String |  |

##### 返回结果：

响应状态码：①200表示成功

②非200表示失败，显示失败的信息描述

##### 返回参数说明：

|  |  |
| --- | --- |
| 返回字段 | 说明 |
| 不处理 |  |

##### 示例：

**请求：**

POST

http://localhost:8080/simple/linux/add/host

…

Content-Type: application/json

…

[

{"ip":"192.168.0.211","user":"root","passwd":"1","cluster":"cluster2"}

]

…

**响应：**

HTTP/1.1 200 OK

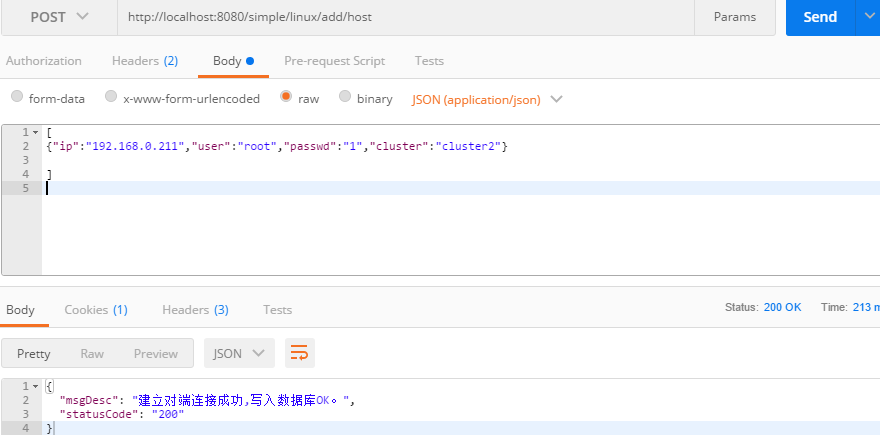
…

{

"msgDesc": "建立对端连接成功,写入数据库OK。",

"statusCode": "200"

}



### 1.2 查询Host

##### 接口地址

GET

<http://localhost:8080/simple/linux/show/host?ip=192.168&page=1&rows=20>

##### 请求参数：

①请求的头行：

Content-Type: application/json

②请求参数：

ip=192.168&page=1&rows=20

请求参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 必选 | 类型 | 默认值 |
| ip | 主机ip | 否 | String |  |
| page | 页码 | 否 | String | 1 |
| rows | 行数 | 否 | String | 20 |

##### 返回结果：

响应状态码：①200表示成功

②非200表示失败，显示失败的信息描述

##### 返回参数说明：

查询Host信息在页面展示ip和cluster两个字段的信息。

|  |  |
| --- | --- |
| 返回字段 | 说明 |
| msgDesc | 返回消息 |
| statusCode | 状态码 |
| totalCount | 总记录数 |
| list |  |
| ip | 主机ip |
| cluster | 集群 |

##### 示例：

**请求：**

GET

http://localhost:8080/simple/linux/show/host?ip=192.168&page=1&rows=20

…

Content-Type: application/json

…

**响应：**

HTTP/1.1 200 OK

…

{

"msgDesc": "OK",

"totalCount": 1,

"list": [

{

"ip": "192.168.0.211",

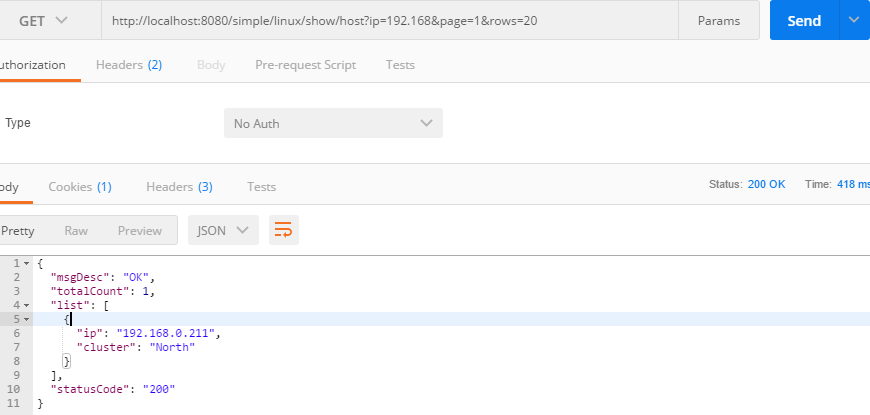
"cluster": "North"

}

],

"statusCode": "200"

}



### 1.3 删除Host

GET

<http://localhost:8080/simple/linux/delete/host?ip=192.168.0.211>

## 2 Users管理

### 2.1 新增User

##### 接口地址

POST

<http://localhost:8080/simple/linux/add/user>

##### 请求参数：

①请求的头行：

Content-Type: application/json

②请求的Body部分(json)：

{

"userName":"qwert",

"passwd":"qwert",

"groups":"mysql",

"cluster":"North",

"description":"password is qwert"

}

请求参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 必选 | 类型 | 默认值 |
| userName | 用户名 | 是 | String |  |
| passwd | 密码 | 是 | String |  |
| groups | 组 | 否 | String |  |
| description | 描述 | 否 | String |  |

##### 返回结果：

响应状态码：①200表示成功

②非200表示失败，显示失败的信息描述

##### 返回参数说明：

|  |  |
| --- | --- |
| 返回字段 | 说明 |
| 不处理 |  |

##### 示例：

**请求：**

POST

[http://localhost:8080/simple/linux/add/user](http://localhost:8080/simple/linux/add/user?cluster=North)

…

Content-Type: application/json

…

{

"userName":"qwert",

"passwd":"qwert",

"groups":"mysql",

"cluster":"North",

"description":"password is qwert"

}

…

**响应：**

HTTP/1.1 200 OK

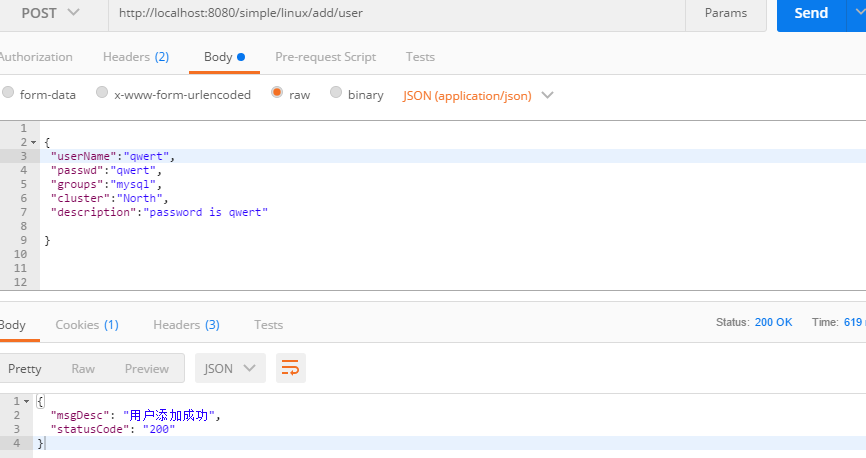
…

{

"msgDesc": "用户添加成功",

"statusCode": "200"

}



### 2.2 查询User

查询User信息，分为查询可以查询数据库里面和服务器里面实际的用户。

##### 接口地址1

查询数据库用户：

GET

<http://localhost:8080/simple/linux/show/dbuser?page=1&rows=20&userName=7&group>=

##### 请求参数：

①请求的头行：

Content-Type: application/json

②请求参数：

page=1&rows=20&userName=7&group=

请求参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 必选 | 类型 | 默认值 |
| userName | 用户名 | 否 | String |  |
| group | 组 | 否 | String |  |
| page | 页码 | 否 | String | 1 |
| rows | 行数 | 否 | String | 20 |

##### 返回结果：

响应状态码：①200表示成功

②非200表示失败，显示失败的信息描述

##### 返回参数说明：

查询用户信息在页面展示userName、groups和description字段的信息。

|  |  |
| --- | --- |
| 返回字段 | 说明 |
| msgDesc | 返回消息 |
| statusCode | 状态码 |
| totalCount | 总记录数 |
| list |  |
| userName | 用户名 |
| groups | 组 |
| description | 描述 |

##### 示例：

查询数据库用户：

**请求：**

GET

<http://localhost:8080/simple/linux/show/dbuser?page=1&rows=20&userName=7&group>=

…

Content-Type: application/json

…

**响应：**

HTTP/1.1 200 OK

…

{

"msgDesc": "查询成功from DB",

"totalCount": 1,

"list": [

{

"userName": "723",

"passwd": "jbyWErT/lQ0=",

"groups": "723",

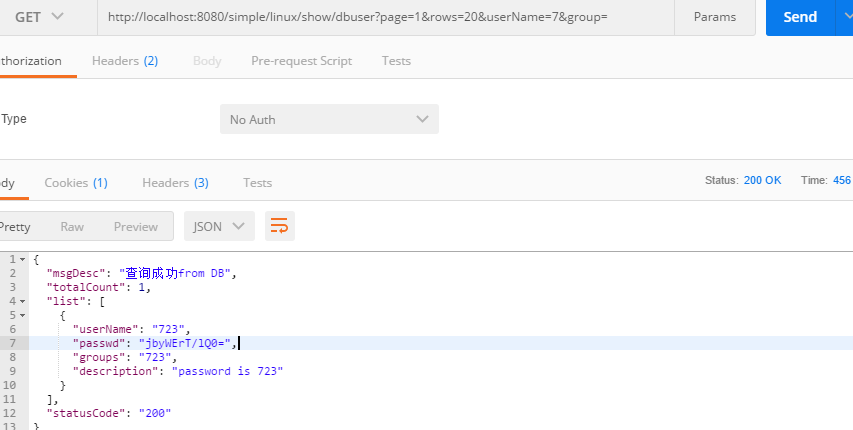
"description": "password is 723"

}

],

"statusCode": "200"

}



##### 接口地址2

查询服务器实际用户：

GET

<http://localhost:8080/simple/linux/show/user?page=1&rows=20&userName=7&group=&ip=192.168.0.211>

##### 请求参数：

①请求的头行：

Content-Type: application/json

②请求参数：

page=1&rows=20&userName=7&group=&ip=192.168.0.211

请求参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 必选 | 类型 | 默认值 |
| userName | 用户名 | 否 | String |  |
| group | 组 | 否 | String |  |
| page | 页码 | 否 | String | 1 |
| rows | 行数 | 否 | String | 20 |
| ip | 服务器ip | 是 | String |  |

##### 返回结果：

响应状态码：①200表示成功

②非200表示失败，显示失败的信息描述

##### 返回参数说明：

查询用户信息在页面展示userName、groups和node字段的信息

|  |  |
| --- | --- |
| 返回字段 | 说明 |
| userName | 用户名 |
| groups | 组 |
| node | 服务器节点 |

##### 示例：

**请求：**

GET

<http://localhost:8080/simple/linux/show/user?page=1&rows=20&userName=7&group=&ip=192.168.0.211>

…

Content-Type: application/json

…

**响应：**

HTTP/1.1 200 OK

…

{

"msgDesc": "查询成功从服务器",

"totalCount": 2,

"list": [

{

"userName": "723",

"groups": "723",

"node": "192.168.0.211"

},

{

"userName": "722",

"groups": "722",

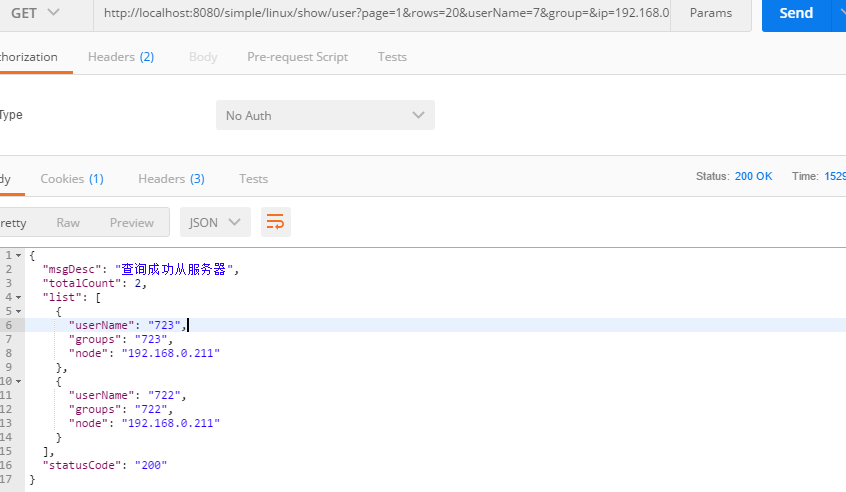
"node": "192.168.0.211"

}

],

"statusCode": "200"

}



### 2.3 删除User

##### 接口地址

POST

<http://localhost:8080/simple/linux/delete/user?userName=723>

##### 请求参数

①请求的头行：

Content-Type: application/json

②请求参数：

userName=722

##### 请求参数说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 必选 | 类型 | 默认值 |
| userName | 用户名 | 是 | String |  |

##### 返回结果

响应状态码：①200表示成功

②非200表示失败，显示失败的信息描述

##### 返回参数说明

|  |  |
| --- | --- |
| 返回字段 | 说明 |
| msgDesc |  |
| statusCode |  |

##### 示例:

请求：

POST

[http://localhost:8080/simple/linux/delete/user?userName=72](http://localhost:8080/simple/linux/delete/user?userName=722)2

…

Content-Type: application/json

…

**响应：**

HTTP/1.1 200 OK

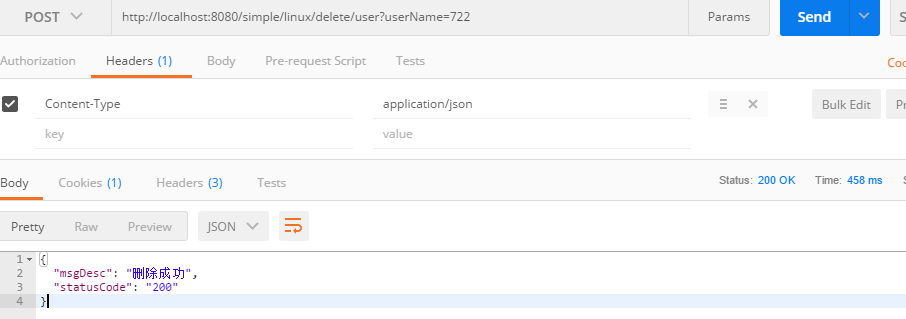
…

{

"msgDesc": "删除成功",

"statusCode": "200"

}



### 2.4 修改User

##### 接口地址

POST

<http://localhost:8080/simple/linux/update/user>

##### 请求参数：

①请求的头行：

Content-Type: application/json

②请求的Body部分(json)：

{

"userName":"722",

"passwd":"722",

"groups":"hadoop",

"description":"password is 722-change-group"

}

请求参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 必选 | 类型 | 默认值 |
| userName | 用户名 | 是 | String |  |
| passwd | 密码 | 是 | String |  |
| groups | 组 | 否 | String |  |
| description | 描述 | 否 | String |  |

##### 返回结果：

响应状态码：①200表示成功

②非200表示失败，显示失败的信息描述

##### 返回参数说明：

|  |  |
| --- | --- |
| 返回字段 | 说明 |
| msgDesc | 返回信息描述 |
| statusCode | 返回状态码 |

##### 示例：

**请求：**

POST

<http://localhost:8080/simple/linux/update/user>

…

Content-Type: application/json

…

{

"userName":"722",

"passwd":"722",

"groups":"hadoop",

"description":"password is 722-change-group"

}

…

**响应：**

HTTP/1.1 200 OK

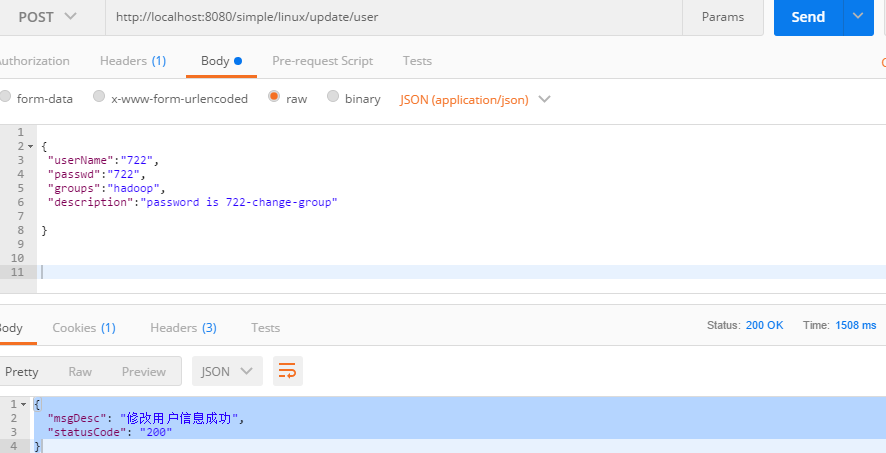
…

{

"msgDesc": "修改用户信息成功",

"statusCode": "200"

}



### 2.5 查看同步User

##### 接口地址

GET

http://localhost:8080/simple/linux/userList?cluster=North

##### 请求参数：

①请求的头行：

Content-Type: application/json

②请求参数：

cluster=North

请求参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 必选 | 类型 | 默认值 |
| cluster | 集群 | 是 | String |  |

##### 返回结果：

响应状态码：①200表示成功

②非200表示失败，显示失败的信息描述

##### 返回参数说明：

查询该集群下同步的用户用于勾选，显示List

|  |  |
| --- | --- |
| 返回字段 | 说明 |
| msgDesc | 返回信息 |
| statusCode | 状态码 |
| list | 该集群下同步的用户list |

##### 示例：

**请求：**

GET

<http://localhost:8080/simple/linux/show/host?ip=192.168&page=1&rows=20>

<http://localhost:8080/simple/linux/userList?cluster=North>

…

Content-Type: application/json

…

**响应：**

HTTP/1.1 200 OK

…

{

"msgDesc": "OK",

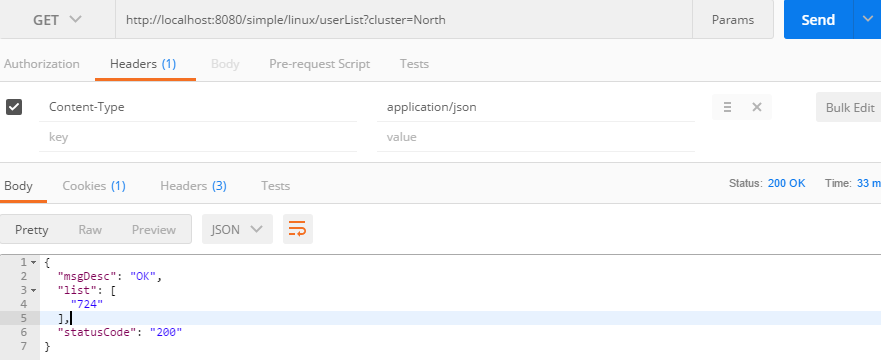
"list": [

"724"

],

"statusCode": "200"

}



### 2.6 同步User

##### 接口地址

POST

<http://localhost:8080/simple/linux/syncUsers?ip=192.168.0.212>

##### 请求参数：

①请求的头行：

Content-Type: application/json

②请求参数：

ip=192.168.0.212

Body部分：

[

"724",

"lolo"

]

请求参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 必选 | 类型 | 默认值 |
| ip | 主机ip | 是 | String |  |
| list | 勾选需要同步的用户 | 是 | List |  |

##### 返回结果：

响应状态码：①200表示成功

②非200表示失败，显示失败的信息描述

##### 返回参数说明：

查询该集群下同步的用户用于勾选，显示List

|  |  |
| --- | --- |
| 返回字段 | 说明 |
| msgDesc | 返回信息 |
| statusCode | 状态码 |

##### 示例：

**请求：**

POST

<http://localhost:8080/simple/linux/syncUsers?ip=192.168.0.212>

…

Content-Type: application/json

Body部分：

[

"724",

"lolo"

]

…

**响应：**

HTTP/1.1 200 OK

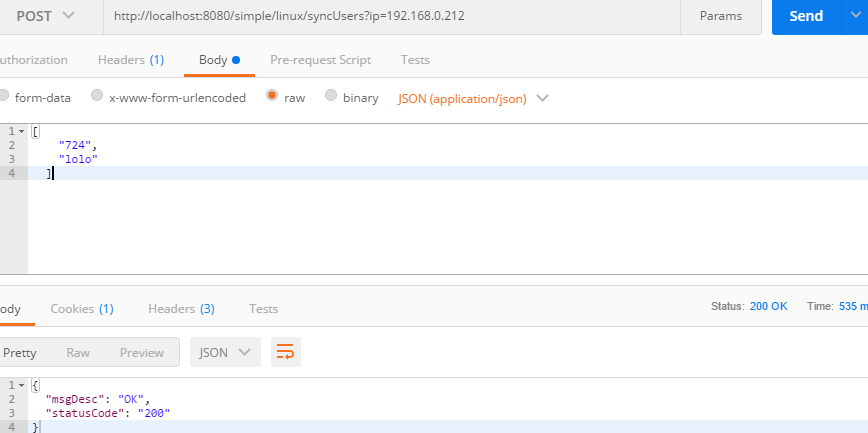
…

[

"724",

"lolo"

]



### 2.7 检查User密码状态

##### 接口地址

GET

http://localhost:8080/simple/linux/check/passwd

##### 请求参数：

①请求的头行：

Content-Type: application/json

②请求参数：

请求参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 必选 | 类型 | 默认值 |
| 无 |  |  |  |  |

##### 返回结果：

响应状态码：①200表示成功

②非200表示失败，显示失败的信息描述

##### 返回参数说明：

|  |  |
| --- | --- |
| 返回字段 | 说明 |
| msgDesc | 返回信息 |
| statusCode | 状态码 |
| list |  |
| userName | 用户名 |
| node | 节点 |

##### 示例：

**请求：**

GET

<http://localhost:8080/simple/linux/check/passwd>

…

Content-Type: application/json

…

**响应：**

HTTP/1.1 200 OK

…

{

"msgDesc": "如下用户的密码发生改变",

"list": [

{

"userName": "724",

"node": "192.168.0.211"

},

{

"userName": "lolo",

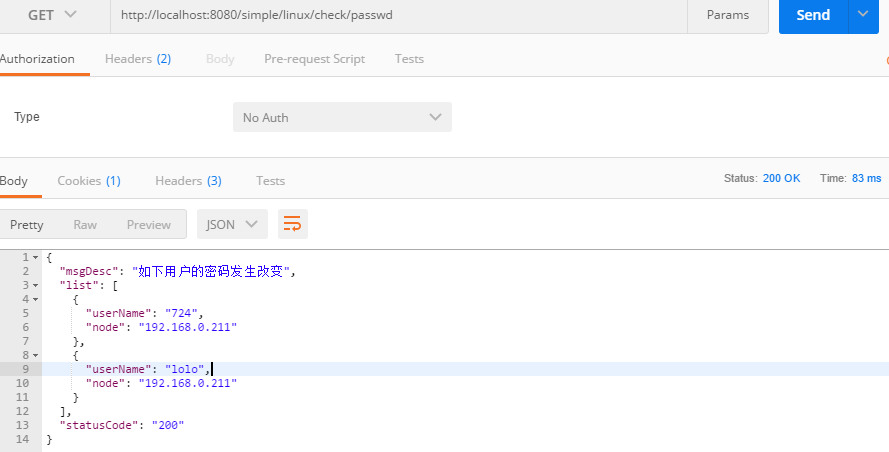
"node": "192.168.0.211"

}

],

"statusCode": "200"

}



### 2.8 User密码重置

##### 接口地址

GET

http://localhost:8080/simple/linux/resetPasswd

##### 请求参数：

①请求的头行：

Content-Type: application/json

②请求参数：

请求参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 必选 | 类型 | 默认值 |
| 无 |  |  |  |  |

##### 返回结果：

响应状态码：①200表示成功

②非200表示失败，显示失败的信息描述

##### 返回参数说明：

|  |  |
| --- | --- |
| 返回字段 | 说明 |
| msgDesc | 返回信息 |
| statusCode | 状态码 |
| list |  |
| userName | 用户名 |
| node | 节点 |

##### 示例：

**请求：**

GET

<http://localhost:8080/simple/linux/resetPasswd>

…

Content-Type: application/json

…

**响应：**

HTTP/1.1 200 OK

…

{

"msgDesc": "重置Linux用户密码成功",

"list": [

{

"userName": "lolo",

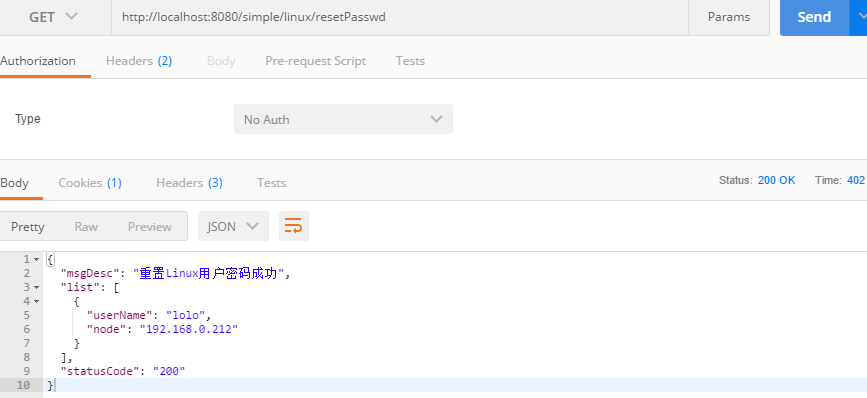
"node": "192.168.0.212"

}

],

"statusCode": "200"

}



## 3 Groups管理

### 3.1 新增Group

##### 接口地址

POST

<http://localhost:8080/simple/linux/add/group>

##### 请求参数：

①请求的头行：

Content-Type: application/json

②请求的Body部分(json)：

{

"groupName":"green",

"cluster":"North",

"description":"you will green"

}

请求参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 必选 | 类型 | 默认值 |
| groupName | 组名 | 是 | String |  |
| cluster | 集群 | 是 | String |  |
| description | 描述 | 否 | String |  |

##### 返回结果：

响应状态码：①200表示成功

②非200表示失败，显示失败的信息描述

##### 返回参数说明：

|  |  |
| --- | --- |
| 返回字段 | 说明 |
| 不处理 |  |

##### 示例：

**请求：**

POST

<http://localhost:8080/simple/linux/add/group>

…

Content-Type: application/json

…

{

"groupName":"green",

"cluster":"North",

"description":"you will green"

}

…

**响应：**

HTTP/1.1 200 OK

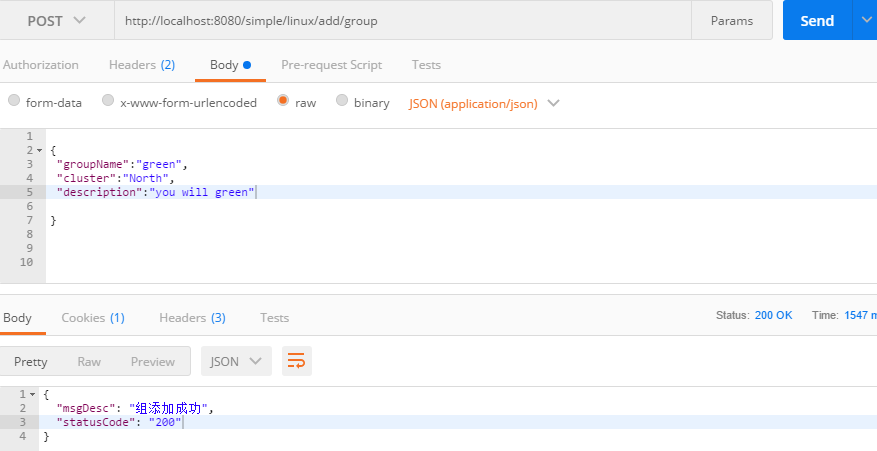
…

{

"msgDesc": "组添加成功",

"statusCode": "200"

}



### 3.2 删除Group

##### 接口地址

POST

<http://localhost:8080/simple/linux/delete/group?groupName=green>

##### 请求参数

①请求的头行：

Content-Type: application/json

②请求参数：

groupName=green

##### 请求参数说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 必选 | 类型 | 默认值 |
| groupName | 组名 | 是 | String |  |

##### 返回结果

响应状态码：①200表示成功

②非200表示失败，显示失败的信息描述

##### 返回参数说明

|  |  |
| --- | --- |
| 返回字段 | 说明 |
| msgDesc |  |
| statusCode |  |

##### 示例:

请求：

POST

<http://localhost:8080/simple/linux/delete/group?groupName>

…

Content-Type: application/json

…

**响应：**

HTTP/1.1 200 OK

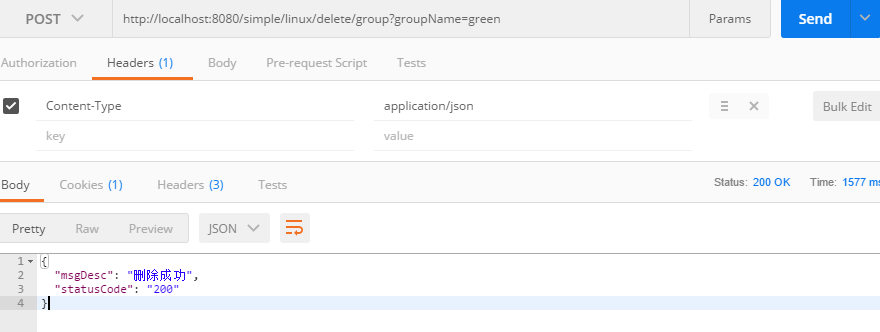
…

{

"msgDesc": "删除成功",

"statusCode": "200"

}



### 3.3 查询Group

##### 接口地址

GET

<http://192.168.0.100:8080/simple/linux/show/group?page=1&rows=20&groupName=g>

##### 请求参数：

①请求的头行：

Content-Type: application/json

②请求参数：

page=1&rows=20&groupName=g

请求参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 必选 | 类型 | 默认值 |
| groupName | 组名 | 否 | String |  |
| page | 页码 | 否 | String | 1 |
| rows | 行数 | 否 | String | 20 |

##### 返回结果：

响应状态码：①200表示成功

②非200表示失败，显示失败的信息描述

##### 返回参数说明：

|  |  |
| --- | --- |
| 返回字段 | 说明 |
| msgDesc | 返回消息 |
| statusCode | 状态码 |
| totalCount | 总记录数 |
| list |  |
| groupName | 组名 |
| cluster | 集群 |
| description | 描述 |

##### 示例：

**请求：**

GET

<http://192.168.0.100:8080/simple/linux/show/group?page=1&rows=20&groupName=g>

…

Content-Type: application/json

…

**响应：**

HTTP/1.1 200 OK

…

{

"msgDesc": "查询成功from DB",

"totalCount": 1,

"list": [

{

"groupName": "green2",

"cluster": "North",

"description": "you will green2"

}

],

"statusCode": "200"

}

