**IDM接口设计文档**

**2014.03**

**修订记录**

|  |  |  |  |
| --- | --- | --- | --- |
| **版本** | **修订内容** | **作者** | **日期** |
| 1.0 | 初稿完成 | 袁欢 | 2014/3/28 |
| 1.1 | 完成全部的接口设计 | 袁欢 | 2014/4/2 |
| 1.2 | 修改部分接口 | 袁欢 | 2014/4/9 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**目 录**

[1 AuthParams 7](#_Toc385504469)

[2 Account 7](#_Toc385504470)

[2.1 createAccount 7](#_Toc385504471)

[2.1.1 接口描述 7](#_Toc385504472)

[2.1.2 请求消息 7](#_Toc385504473)

[2.1.3 响应消息 8](#_Toc385504474)

[2.2 deleteAccount 9](#_Toc385504475)

[2.2.1 接口描述 9](#_Toc385504476)

[2.2.2 请求消息 9](#_Toc385504477)

[2.2.3 响应消息 9](#_Toc385504478)

[3 User 10](#_Toc385504479)

[3.1 createUser 10](#_Toc385504480)

[3.1.1 接口描述 10](#_Toc385504481)

[3.1.2 请求消息 10](#_Toc385504482)

[3.1.3 响应消息 11](#_Toc385504483)

[3.2 getUser 12](#_Toc385504484)

[3.2.1 接口描述 12](#_Toc385504485)

[3.2.2 请求消息 12](#_Toc385504486)

[3.2.3 响应消息 12](#_Toc385504487)

[3.3 listUsers 13](#_Toc385504488)

[3.3.1 接口描述 13](#_Toc385504489)

[3.3.2 请求消息 13](#_Toc385504490)

[3.3.3 响应消息 14](#_Toc385504491)

[3.4 deleteUser 15](#_Toc385504492)

[3.4.1 接口描述 15](#_Toc385504493)

[3.4.2 请求消息 15](#_Toc385504494)

[3.4.3 响应消息 15](#_Toc385504495)

[3.5 updateUser 16](#_Toc385504496)

[3.5.1 接口描述 16](#_Toc385504497)

[3.5.2 请求消息 16](#_Toc385504498)

[3.5.3 响应消息 17](#_Toc385504499)

[3.6 changePassword 18](#_Toc385504500)

[3.6.1 接口描述 18](#_Toc385504501)

[3.6.2 请求消息 18](#_Toc385504502)

[3.6.3 响应消息 18](#_Toc385504503)

[4 Group 19](#_Toc385504504)

[4.1 createGroup 19](#_Toc385504505)

[4.1.1 接口描述 19](#_Toc385504506)

[4.1.2 请求消息 19](#_Toc385504507)

[4.1.3 响应消息 20](#_Toc385504508)

[4.2 getGroup 21](#_Toc385504509)

[4.2.1 接口描述 21](#_Toc385504510)

[4.2.2 请求消息 21](#_Toc385504511)

[4.2.3 响应消息 21](#_Toc385504512)

[4.3 listGroups 22](#_Toc385504513)

[4.3.1 接口描述 22](#_Toc385504514)

[4.3.2 请求消息 22](#_Toc385504515)

[4.3.3 响应消息 23](#_Toc385504516)

[4.4 deleteGroup 24](#_Toc385504517)

[4.4.1 接口描述 24](#_Toc385504518)

[4.4.2 请求消息 24](#_Toc385504519)

[4.4.3 响应消息 24](#_Toc385504520)

[4.5 updateGroup 25](#_Toc385504521)

[4.5.1 接口描述 25](#_Toc385504522)

[4.5.2 请求消息 25](#_Toc385504523)

[4.5.3 响应消息 26](#_Toc385504524)

[4.6 addUserToGroup 26](#_Toc385504525)

[4.6.1 接口描述 26](#_Toc385504526)

[4.6.2 请求消息 26](#_Toc385504527)

[4.6.3 响应消息 27](#_Toc385504528)

[4.7 removeUserFromGroup 28](#_Toc385504529)

[4.7.1 接口描述 28](#_Toc385504530)

[4.7.2 请求消息 28](#_Toc385504531)

[4.7.3 响应消息 28](#_Toc385504532)

[5 AccessKey 29](#_Toc385504533)

[5.1 createAccessKey 29](#_Toc385504534)

[5.1.1 接口描述 29](#_Toc385504535)

[5.1.2 请求消息 29](#_Toc385504536)

[5.1.3 响应消息 30](#_Toc385504537)

[5.2 deleteAccessKey 31](#_Toc385504538)

[5.2.1 接口描述 31](#_Toc385504539)

[5.2.2 请求消息 31](#_Toc385504540)

[5.2.3 响应消息 31](#_Toc385504541)

[5.3 updateAccessKey 32](#_Toc385504542)

[5.3.1 接口描述 32](#_Toc385504543)

[5.3.2 请求消息 32](#_Toc385504544)

[5.3.3 响应消息 33](#_Toc385504545)

[5.4 listAccessKeys 34](#_Toc385504546)

[5.4.1 接口描述 34](#_Toc385504547)

[5.4.2 请求消息 34](#_Toc385504548)

[5.4.3 响应消息 34](#_Toc385504549)

[6 Policy 35](#_Toc385504550)

[6.1 createUserPolicy 35](#_Toc385504551)

[6.1.1 接口描述 35](#_Toc385504552)

[6.1.2 请求消息 35](#_Toc385504553)

[6.1.3 响应消息 36](#_Toc385504554)

[6.2 updateUserPolicy 37](#_Toc385504555)

[6.2.1 接口描述 37](#_Toc385504556)

[6.2.2 请求消息 37](#_Toc385504557)

[6.2.3 响应消息 38](#_Toc385504558)

[6.3 getUserPolicy 39](#_Toc385504559)

[6.3.1 接口描述 39](#_Toc385504560)

[6.3.2 请求消息 39](#_Toc385504561)

[6.3.3 响应消息 39](#_Toc385504562)

[6.4 listUserPolicys 40](#_Toc385504563)

[6.4.1 接口描述 40](#_Toc385504564)

[6.4.2 请求消息 40](#_Toc385504565)

[6.4.3 响应消息 41](#_Toc385504566)

[6.5 deleteUserPolicy 42](#_Toc385504567)

[6.5.1 接口描述 42](#_Toc385504568)

[6.5.2 请求消息 42](#_Toc385504569)

[6.5.3 响应消息 43](#_Toc385504570)

[6.6 createGroupPolicy 43](#_Toc385504571)

[6.6.1 接口描述 43](#_Toc385504572)

[6.6.2 请求消息 43](#_Toc385504573)

[6.6.3 响应消息 44](#_Toc385504574)

[6.7 updateGroupPolicy 45](#_Toc385504575)

[6.7.1 接口描述 45](#_Toc385504576)

[6.7.2 请求消息 45](#_Toc385504577)

[6.7.3 响应消息 46](#_Toc385504578)

[6.8 getGroupPolicy 46](#_Toc385504579)

[6.8.1 接口描述 46](#_Toc385504580)

[6.8.2 请求消息 47](#_Toc385504581)

[6.8.3 响应消息 47](#_Toc385504582)

[6.9 listGroupPolicys 48](#_Toc385504583)

[6.9.1 接口描述 48](#_Toc385504584)

[6.9.2 请求消息 48](#_Toc385504585)

[6.9.3 响应消息 49](#_Toc385504586)

[6.10 deleteGroupPolicy 50](#_Toc385504587)

[6.10.1 接口描述 50](#_Toc385504588)

[6.10.2 请求消息 50](#_Toc385504589)

[6.10.3 响应消息 50](#_Toc385504590)

[7 Authentication 51](#_Toc385504591)

[7.1 authenticateRequest 51](#_Toc385504592)

[7.1.1 接口描述 51](#_Toc385504593)

[7.1.2 请求消息 51](#_Toc385504594)

[7.1.3 响应消息 52](#_Toc385504595)

[8 HTTP接口状态码和错误信息定义 53](#_Toc385504596)

[8.1 IDM系统错误码定义 53](#_Toc385504597)

[8.2 HTTP状态码与IDM错误码对应关系 55](#_Toc385504598)

[8.3 错误信息格式定义 59](#_Toc385504599)

# AuthParams

定义用于用户请求认证的参数字段，在所有接口的请求参数中都必须携带这些参数。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| twsAccessKeyId | 用于接口认证的AccessKey ID | String | 必选 |  |
| signatureMethod | 数字签名算法 | String | 必选 | 目前只支持HmacSHA256 |
| signatureVersion | 数字签名版本 | String | 必选 | 目前只支持1 |
| signature | 数字签名 | String | 必选 |  |
| timestamp | 时间戳 | String | 必选 | 采用UTC时间。格式：  yyyy-MM-dd'T'HH:mm:ss'Z'，例如：2014-03-28T14:48:31Z |

# Account

用户操作接口只能由管理员账户来完成，其他用户无权操作。

## createAccount

### 接口描述

创建管理员账户。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/createAccount

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| accountName | 账户名 | String | 必选 |  |
| password | 密码 | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/createAccount?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&accountName=Bob&password=123456

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| accountId | 账户ID | String | 必选 |  |
| accountName | 用户名 | String | 必选 |  |
| createDate | 创建时间 | String | 必选 |  |
| defaultAccessKeyId | 默认AccessKeyID | String | 必选 |  |
| defaultSecretAccessKey | 默认SecretAccessKey | String | 必选 |  |

* **消息样例**

成功：

{"accountId":"ae38844d-0733-40ce-b846-e8d79d61871a","accountName":"Bob","createDate":"2014-02-02T10:00:11Z"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## deleteAccount

### 接口描述

删除账户。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/deleteUser

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| accountId | 账户名 | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/deleteUser?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&accountId=ae38844d-0733-40ce-b846-e8d79d61871a

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| result | 删除结果 | String | 必选 | 取值为Success |

* **消息样例**

成功：

{"reuslt":"Success"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

# User

用户操作接口只能由管理员账户来完成，其他用户无权操作。

## createUser

### 接口描述

为管理员账户创建一个普通用户。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/createUser

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userName | 用户名 | String | 必选 |  |
| password | 密码 | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/createUser?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&userName=Bob&password=123456

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userId | 用户ID | String | 必选 |  |
| userName | 用户名 | String | 必选 |  |
| accountId | 所属的账户ID | String | 必选 |  |
| createDate | 创建时间 | String | 必选 |  |

* **消息样例**

成功：

{"userId":"ae38844d-0733-40ce-b846-e8d79d61871a","userName":"Bob","accountId":"10000000","createDate":"2014-02-02T10:00:11Z"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## getUser

### 接口描述

查询用户信息。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/getUser

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userId | 用户名 | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/getUser?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&userId=ae38844d-0733-40ce-b846-e8d79d61871a

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userId | 用户ID | String | 必选 |  |
| userName | 用户名 | String | 必选 |  |
| accountId | 所属账户ID | String | 必选 |  |
| createDate | 电子邮箱 | String | 必选 |  |

* **消息样例**

成功：

{"userId":"ae38844d-0733-40ce-b846-e8d79d61871a","userName":"Bob","accountId":"10000000","createDate":"2014-03-23T12:23:42Z"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## listUsers

### 接口描述

查询所有用户。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/listUsers

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| 无 |  |  |  |  |

* **消息样例**

http://127.0.0.1:8080/idm/listUsers?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

List<User>，其中User的字段如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userId | 用户ID | String | 必选 |  |
| userName | 用户名 | String | 必选 |  |
| accountId | 所属的账户ID | String | 必选 |  |
| createDate | 创建时间 | String | 必选 |  |

* **消息样例**

成功：

[{"userId":"ae38844d-0733-40ce-b846-e8d79d61871a","userName":"Bob","accountId":"10000000","createDate":"2014-03-23T12:23:42Z"},{"userId":"10000002","userName":"Tom","accountId":"10000000","createDate":"2014-03-23T12:24:42Z"}]

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## deleteUser

### 接口描述

删除用户信息。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/deleteUser

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userId | 用户名 | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/deleteUser?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&userId=10000002

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| result | 删除结果 | String | 必选 | 取值为Success |

* **消息样例**

成功：

{"reuslt":"Success"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## updateUser

### 接口描述

修改用户信息。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/updateUser

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userId | 用户ID | String | 必选 |  |
| userName | 用户名 | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/updateUser?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&userId=ae38844d-0733-40ce-b846-e8d79d61871a&userName=NewBob

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userId | 用户ID | String | 必选 |  |
| userName | 用户名 | String | 必选 |  |
| accountId | 所属账户ID | String | 必选 |  |
| createDate | 电子邮箱 | String | 必选 |  |

* **消息样例**

成功：

{"userId":"ae38844d-0733-40ce-b846-e8d79d61871a","userName":"Bob","accountId":"10000000","createDate":"2014-03-23T12:23:42Z"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## changePassword

### 接口描述

修改用户密码。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/changePassword

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userId | 用户ID | String | 必选 |  |
| oldPassword | 旧密码 | String | 必选 |  |
| newPassword | 新密码 | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/updateUser?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&userId=ae38844d-0733-40ce-b846-e8d79d61871a&oldPassword=123456&newPassword=888888

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| result | 删除结果 | String | 必选 | 取值为Success |

* **消息样例**

成功：

{"reuslt":"Success"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

# Group

组操作接口只能由管理员账户来完成，其他用户无权操作。

## createGroup

### 接口描述

创建一个组。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/createGroup

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| groupName | 组名 | String | 必选 |  |
| description | 描述 | String | 可选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/createGroup?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&groupName=GroupName\_1395989992775&description=test

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| groupId | 组ID | String | 必选 |  |
| groupName | 组名 | String | 必选 |  |
| createDate | 创建时间 | String | 必选 |  |
| description | 描述 | String | 可选 |  |

* **消息样例**

成功：

{"groupId":"ae38844d-0733-40ce-b846-e8d79d61871a","groupName":"GroupName\_1395989992775","description":"test","createDate":"2014-03-26T13:34:21Z"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## getGroup

### 接口描述

查询组信息。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/getGroup

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| groupId | 组ID | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/getGroup?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&groupId=ae38844d-0733-40ce-b846-e8d79d61871a

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| groupId | 组ID | String | 必选 |  |
| groupName | 组名 | String | 必选 |  |
| description | 描述 | String | 可选 |  |
| createDate | 创建时间 | String | 必选 |  |

* **消息样例**

成功：

{"groupId":"ae38844d-0733-40ce-b846-e8d79d61871a","groupName":"GroupName\_1395989992775","description":"test","createDate":"2014-03-26T13:34:21Z"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## listGroups

### 接口描述

查询组列表。如果指定用户名，则查询该用户所属于的组列表。如果不指定用户名，则查询所有组列表。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/listGroups

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userId | 用户ID | String | 可选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/listGroupsForUser?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&userId=ae38844d-0733-40ce-b846-e8d79d61871a

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

List<Group>，其中Group的字段如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| groupId | 组ID | String | 必选 |  |
| groupName | 组名 | String | 必选 |  |
| description | 描述 | String | 可选 |  |
| createDate | 创建时间 | String | 必选 |  |

* **消息样例**

成功：

[{"groupId":"ae38844d-0733-40ce-b846-e8d79d61871a","groupName":"GroupName\_1395989992775","description":"test","createDate":"2014-03-26T13:34:21Z"},{"groupId":"10000002","groupName":"GroupName\_1395989992777","description":"test","createDate":"2014-03-27T13:34:21Z"}]

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## deleteGroup

### 接口描述

删除组信息。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/deleteGroup

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| groupId | 组名 | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/deleteGroup?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&groupId=10000002

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| result | 删除结果 | String | 必选 | 取值为Success |

* **消息样例**

成功：

{"reuslt":"Success"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## updateGroup

### 接口描述

修改组信息。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/updateGroup

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| groupId | 组名 | String | 必选 |  |
| groupName | 新组名 | String | 可选 | groupNamehe和description至少有一个字段不为空。 |
| description | 描述 | String | 可选 |

* **消息样例**

http://127.0.0.1:8080/idm/updateGroup?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOigroupId=ae38844d-0733-40ce-b846-e8d79d61871a&&groupName=test1

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| groupId | 组ID | String | 必选 |  |
| groupName | 组名 | String | 必选 |  |
| description | 描述 | String | 可选 |  |
| createDate | 创建时间 | String | 必选 |  |

* **消息样例**

成功：

{"groupId":"ae38844d-0733-40ce-b846-e8d79d61871a","groupName":"GroupName\_1395989992775","description":"test","createDate":"2014-03-26T13:34:21Z"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## addUserToGroup

### 接口描述

将一个用户添加到组。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/addUserToGroup

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userId | 用户ID | String | 必选 |  |
| groupId | 组ID | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/addUserToGroup?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&userId=ae38844d-0733-40ce-b846-e8d79d61871a&groupId=10000002

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| result | 删除结果 | String | 必选 | 取值为Success |

* **消息样例**

成功：

{"reuslt":"Success"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## removeUserFromGroup

### 接口描述

将一个用户从组中删除。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/removeUserFromGroup

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userId | 用户ID | String | 必选 |  |
| groupId | 组ID | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/removeUserFromGroup?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&userId=ae38844d-0733-40ce-b846-e8d79d61871a&groupId=10000002

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| result | 删除结果 | String | 必选 | 取值为Success |

* **消息样例**

成功：

{"reuslt":"Success"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

# AccessKey

## createAccessKey

### 接口描述

为用户创建一对AccessKey。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/createAccessKey

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userId | 用户ID | String | 可选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/createAccessKey?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&userId=ae38844d-0733-40ce-b846-e8d79d61871a

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| accessKeyId | AccessKeyID | String | 必选 |  |
| secretAccessKey | SecretAccessKey | String | 必选 |  |
| status | 状态 | String | 必选 |  |
| ownerId | 所属用户ID | String | 必选 |  |
| createDate | 创建日期 | String | 必选 |  |

* **消息样例**

成功：

{"accessKeyId":"TgJDCBcuDUp50N0bdf0t22u2w0yEx5jo","createDate":"2014-03-28T16:47:01Z","secretAccessKey":"tquTwytPrEqZMrvf64BG5QvtdNMr5VuHcKlczotgvuK2APGM","status":"active","ownerId":"ae38844d-0733-40ce-b846-e8d79d61871a"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## deleteAccessKey

### 接口描述

删除一对AccessKey，如果没有指定UserName的话则会根据AccessKey的属主关系来删除。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/deleteAccessKey

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| accessKeyId | AccessKey的ID | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/deleteAccessKey?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&accessKeyId=TgJDCBcuDUp50N0bdf0t22u2w0yEx5jo

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| result | 删除结果 | String | 必选 | 取值为Success |

* **消息样例**

成功：

{"reuslt":"Success"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## updateAccessKey

### 接口描述

更新AccessKey的状态。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/updateAccessKey

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| accessKeyId | AccessKey的ID | String | 必选 |  |
| status | 状态 | String | 必选 | 取值：active、inactive |

* **消息样例**

http://127.0.0.1:8080/idm/updateAccessKey?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&accessKeyId=TgJDCBcuDUp50N0bdf0t22u2w0yEx5jo&status=active

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| accessKeyId | AccessKeyID | String | 必选 |  |
| status | 状态 | String | 必选 |  |
| ownerId | 所属用户ID | String | 必选 |  |
| createDate | 创建日期 | String | 必选 |  |

* **消息样例**

成功：

{"accessKeyId":"TgJDCBcuDUp50N0bdf0t22u2w0yEx5jo","createDate":"2014-03-28T16:47:01Z","status":"active","ownerId":"ae38844d-0733-40ce-b846-e8d79d61871a"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## listAccessKeys

### 接口描述

查询用户的所有AccessKey信息。如果没有指定UserName的话，则查询当前用户的所有AccessKey。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/listAccessKeys

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userId | 用户ID | String | 可选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/listAccessKeys?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YYAeCU4cOi&userId=ae38844d-0733-40ce-b846-e8d79d61871a

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

List<AccessKeyMetadata>，其中AccessKeyMetadata的字段如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| accessKeyId | AccessKeyID | String | 必选 |  |
| status | 状态 | String | 必选 |  |
| ownerId | 所属用户ID | String | 必选 |  |
| createDate | 创建日期 | String | 必选 |  |

* **消息样例**

成功：

[{"accessKeyId":"TgJDCBcuDUp50N0bdf0t22u2w0yEx5jo","createDate":"2014-03-28T16:47:01Z","status":"active","userId":"ae38844d-0733-40ce-b846-e8d79d61871a"},{"accessKeyId":"TgJDCBcuDUp50N0bdf0t22u2w0yEx54f","createDate":"2014-03-28T16:48:01Z","status":"active","ownerId":"ae38844d-0733-40ce-b846-e8d79d61871a"}]

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

# Policy

## createUserPolicy

### 接口描述

为用户新增访问控制策略文档。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

POST

* **服务地址URI**

http://{SERVER\_HOST}/idm/createUserPolicy

* **消息格式**

采用HTTP POST请求传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userId | 用户ID | String | 必选 |  |
| policyName | 策略名 | String | 必选 |  |
| policyDocument | 策略文档 | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/createUserPolicy

POST Data:

|  |
| --- |
| {"twsAccessKeyId":"5O7nfOB3yLeQtJZe6Woi93YYAeCU4cOi", "policyName":"newPolicy", "userId": "ae38844d-0733-40ce-b846-e8d79d61871a", "policyDocument": [{"effect":"allow", "resource":"\*:\*"}]} |

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| policyId | 策略名ID | String | 必选 |  |
| policyName | 策略名称 | String | 必选 |  |
| userId | 用户名 | String | 必选 |  |
| policyDocument | 策略文档 | String | 必选 |  |

* **消息样例**

成功：

{"policyId":"db6ec1d6-196e-4cd1-8376-e9e5eeed5717","policyName":"newPolicy", "userId": "ead25b07-d04d-4aeb-9e3c-c4d0a1e4d1d1", "policyDocument": [{"effect":"allow", “resource”:”\*:\*”}]}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## updateUserPolicy

### 接口描述

为用户修改访问控制策略文档。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

POST

* **服务地址URI**

http://{SERVER\_HOST}/idm/updateUserPolicy

* **消息格式**

采用HTTP POST请求传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userId | 用户ID | String | 必选 |  |
| policyId | 策略ID | String | 必选 |  |
| policyName | 策略名 | String | 可选 | policyName和policyDocument至少一个不为空。 |
| policyDocument | 策略文档 | String | 可选 |

* **消息样例**

http://127.0.0.1:8080/idm/updateUserPolicy

POST Data:

|  |
| --- |
| {"twsAccessKeyId":"5O7nfOB3yLeQtJZe6Woi93YYAeCU4cOi", "policyName":"newPolicy", "userId": "ae38844d-0733-40ce-b846-e8d79d61871a", "policyId": "d638844d-0733-40ce-b846-e8d79d61871a", "policyDocument": [{"effect":"allow", “resource”:”\*:\*”}]} |

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| result | 删除结果 | String | 必选 | 取值为Success |

* **消息样例**

成功：

{"reuslt":"Success"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## getUserPolicy

### 接口描述

查询用户的访问控制策略文档。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/getUserPolicy

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| policyId | 策略ID | String | 必选 |  |
| userId | 用户ID | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/getUserPolicy?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&policyId=2f65844d-0733-40ce-b846-e8d79d61871a&userId=ae38844d-0733-40ce-b846-e8d79d61871a

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| policyId | 策略名ID | String | 必选 |  |
| policyName | 策略名称 | String | 必选 |  |
| userId | 用户名 | String | 必选 |  |
| policyDocument | 策略文档 | String | 必选 |  |

* **消息样例**

成功：

{"policyId":"db6ec1d6-196e-4cd1-8376-e9e5eeed5717","policyName":"newPolicy", "userId": "ead25b07-d04d-4aeb-9e3c-c4d0a1e4d1d1", "policyDocument": [{"effect":"allow", “resource”:”\*:\*”}]}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## listUserPolicys

### 接口描述

查询用户的访问控制策略列表。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/listUserPolicys

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| userId | 用户ID | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/listUserPolicys?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&userId=ae38844d-0733-40ce-b846-e8d79d61871a

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

List<UserPolicy>对象，其中UserPolicy的字段如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| policyId | 策略名ID | String | 必选 |  |
| policyName | 策略名称 | String | 必选 |  |
| userId | 用户名 | String | 必选 |  |
| policyDocument | 策略文档 | String | 必选 |  |

* **消息样例**

成功：

[{"policyId": "2d15844d-0733-40ce-b846-e8d79d61871a","policyName":"newPolicy01", "userId": "ae38844d-0733-40ce-b846-e8d79d61871a", "policyDocument": [{"effect":"allow", “resource”:”\*:\*”}]},{"policyId": "6fv4844d-0733-40ce-b846-e8d79d61871a","policyName":"newPolicy02", "userId": "ae38844d-0733-40ce-b846-e8d79d61871a", "policyDocument": [{"effect":"allow", “resource”:”\*:\*”}]}]

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## deleteUserPolicy

### 接口描述

删除用户的访问控制策略文档。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/deleteUserPolicy

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| policyId | 策略ID | String | 必选 |  |
| userId | 用户ID | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/deleteUserPolicy?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&policyId=2f65844d-0733-40ce-b846-e8d79d61871a&userd=ae38844d-0733-40ce-b846-e8d79d61871a

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| result | 删除结果 | String | 必选 | 取值为Success |

* **消息样例**

成功：

{"reuslt":"Success"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## createGroupPolicy

### 接口描述

为组新增访问控制策略文档。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

POST

* **服务地址URI**

http://{SERVER\_HOST}/idm/createGroupPolicy

* **消息格式**

采用HTTP POST请求传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| groupId | 组ID | String | 必选 |  |
| policyName | 策略名 | String | 必选 |  |
| policyDocument | 策略文档 | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/createGroupPolicy

POST Data:

|  |
| --- |
| {"twsAccessKeyId":"5O7nfOB3yLeQtJZe6Woi93YYAeCU4cOi", "policyName":"newPolicy", "groupId": "ae38844d-0733-40ce-b846-e8d79d61871a", "policyDocument": [{"effect":"allow", “resource”:”\*:\*”}]} |

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| policyId | 策略名ID | String | 必选 |  |
| policyName | 策略名称 | String | 必选 |  |
| groupId | 组名 | String | 必选 |  |
| policyDocument | 策略文档 | String | 必选 |  |

* **消息样例**

成功：

{"policyId":"db6ec1d6-196e-4cd1-8376-e9e5eeed5717","policyName":"newPolicy", "groupId": "ead25b07-d04d-4aeb-9e3c-c4d0a1e4d1d1", "policyDocument": [{"effect":"allow", "resource":"\*:\*"}]}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## updateGroupPolicy

### 接口描述

为组修改访问控制策略文档。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

POST

* **服务地址URI**

http://{SERVER\_HOST}/idm/updateGroupPolicy

* **消息格式**

采用HTTP POST请求传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| groupId | 组ID | String | 必选 |  |
| policyId | 策略ID | String | 必选 |  |
| policyName | 策略名 | String | 必选 |  |
| policyDocument | 策略文档 | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/updateGroupPolicy

POST Data:

|  |
| --- |
| {"twsAccessKeyId":"5O7nfOB3yLeQtJZe6Woi93YYAeCU4cOi", "policyName":"newPolicy", "groupId": "ae38844d-0733-40ce-b846-e8d79d61871a", "policyId": "d638844d-0733-40ce-b846-e8d79d61871a", "policyDocument": [{"effect":"allow", “resource”:”\*:\*”}]} |

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| result | 删除结果 | String | 必选 | 取值为Success |

* **消息样例**

成功：

{"reuslt":"Success"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## getGroupPolicy

### 接口描述

查询组的访问控制策略文档。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/getGroupPolicy

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| policyId | 策略ID | String | 必选 |  |
| groupId | 组ID | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/getGroupPolicy?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&policyId=2f65844d-0733-40ce-b846-e8d79d61871a&groupd=ae38844d-0733-40ce-b846-e8d79d61871a

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| policyId | 策略名ID | String | 必选 |  |
| policyName | 策略名称 | String | 必选 |  |
| groupId | 组名 | String | 必选 |  |
| policyDocument | 策略文档 | String | 必选 |  |

* **消息样例**

成功：

{"policyId":"db6ec1d6-196e-4cd1-8376-e9e5eeed5717","policyId":"newPolicy", "groupId": "dad25b07-d04d-4aeb-9e3c-c4d0a1e4d1d1", "policyDocument": [{"effect":"allow", “resource”:”\*:\*”}]}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## listGroupPolicys

### 接口描述

查询组的访问控制策略列表。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/listGroupPolicys

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| groupId | 组ID | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/listGroupPolicys?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&groupId=ae38844d-0733-40ce-b846-e8d79d61871a

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

List<GroupPolicy>对象，其中GroupPolicy的字段如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| policyId | 策略名ID | String | 必选 |  |
| policyName | 策略名称 | String | 必选 |  |
| groupId | 组名 | String | 必选 |  |
| policyDocument | 策略文档 | String | 必选 |  |

* **消息样例**

成功：

[{"policyId": "2d15844d-0733-40ce-b846-e8d79d61871a","policyName":"newPolicy01", "groupId": "ae38844d-0733-40ce-b846-e8d79d61871a", "policyDocument": [{"effect":"allow", “resource”:”\*:\*”}]},{"policyId": "6fv4844d-0733-40ce-b846-e8d79d61871a","policyName":"newPolicy02", "groupId": "ae38844d-0733-40ce-b846-e8d79d61871a", "policyDocument": [{"effect":"allow", “resource”:”\*:\*”}]}]

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

## deleteGroupPolicy

### 接口描述

删除组的访问控制策略文档。

### 请求消息

* **消息流向**

请求客户端 => IDM

* **HTTP请求方式**

GET

* **服务地址URI**

http://{SERVER\_HOST}/idm/deleteGroupPolicy

* **消息格式**

采用HTTP GET请求的QueryString方式传递参数。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| policyId | 策略ID | String | 必选 |  |
| groupId | 组ID | String | 必选 |  |

* **消息样例**

http://127.0.0.1:8080/idm/deleteGroupPolicy?twsAccessKeyId=5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi&policyId=2f65844d-0733-40ce-b846-e8d79d61871a&groupd=ae38844d-0733-40ce-b846-e8d79d61871a

### 响应消息

* **消息流向**

IDM => 请求客户端

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| result | 删除结果 | String | 必选 | 取值为Success |

* **消息样例**

成功：

{"reuslt":"Success"}

失败：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

# Authentication

## authenticate

### 接口描述

用于Web Service API Server对用户进行请求认证。

### 请求消息

* **消息流向**

Cloud API Server => IDM

* **HTTP请求方式**

POST

* **服务地址URI**

http://{SERVER\_HOST}/idm/authenticate

* **消息格式**

JSON

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| toBeSignedString | 客户端待签名字符串 | String | 必选 |  |
| policyResource | 策略资源信息 | String | 必选 | 格式为：{服务类型}：{API接口} |
| reqSignature | 用户客户端请求的数字签名 | String | 必选 |  |
| reqAccessKeyId | AccessKey ID | String | 必选 |  |
| reqSignatureMethod | 用户客户端请求的数字签名算法 | String | 必选 | 目前只支持HmacSHA256 |
| reqSignatureVersion | 用户客户端请求的数字签名版本 | String | 必选 | 目前只支持1 |
| reqTimestamp | 用户客户端请求的时间戳 | String | 必选 |  |
| reqClientIP | 用户客户端IP | String | 必选 |  |

* **消息样例**

URL: <http://127.0.0.1:8080/idm/authenticate>

POST Data:

{"twsAccessKeyId":"5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi","timestamp":"2014-04-18T15:03:00Z","reqTimestamp":"2014-04-18T15:03:00Z","signatureMethod":"HmacSHA256","signatureVersion":"1","signature":"15sdaf51sdfFhwgsj3gd87sghda7dg87asgdiu23h==","reqAccessKeyId":"5O7nfOB3yLeQtJZe6Woi93YAeCUX4cOi","reqSignatureMethod":"HmacSHA256","reqSignatureVersion":"1","reqSignature":"7G93hsd7GK3gGFhwgsj3gd87sghda7dg87asgdiu23h==","policyResource":"transcoding:\*","reqClientIP":"54.254.35.245","toBeSignedString":"54.254.35.245"}

### 响应消息

* **消息流向**

IDM => Cloud API Server

* **消息格式**

采用Json格式传输消息。

* **消息参数字段**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| result | 删除结果 | String | 必选 | 取值为Success |

* **消息样例**

认证成功：

{"reuslt":"Success"}

认证失败：

{"reuslt":"AuthFailure", "reusltMessage":"The provided credentials could not be validated."}

接口访问错误：

{"ErrorCode": "AuthFailure", "ErrorMessage": "The provided credentials could not be validated."}

# 实体定义

## Policy

### policyDocument

JSONArray格式的字符串，每一个JSONObject定义为PolicyDocumentType。

PolicyDocumentType定义如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **字段类型** | **是否必选** | **备注** |
| effect | 用户ID | String | 必选 | 取值范围：allow/deny |
| resource | 策略名 | String | 必选 | 格式：  *[Service\_Name]*:*[API\_Name]*  支持通配符\*。  样例：  1. \*:\*  2. transcode:\*  3. transcode:transcoder |

目前支持的Service及其API如下：

|  |  |
| --- | --- |
| Service | API |
| transcode | transcoder |
| logo |
| snapshots |
| clipper |
| insertion |
| cameratake | cameratake |

policyDocument样例：

1. [{"effect":"allow", "resource":"\*:\*"}]

允许访问所有Service的所有接口。

2. [{"effect":"allow", "resource":"transcode:snapshots"},{"effect":"allow", "resource":"cameratake:\*"}]

允许访问transcode服务下面的snapshots接口和cameratake服务的所有接口。

3. [{"effect":"allow", "resource":"transcode:\*"},{"effect":"deny", "resource":"transcode:logo"}]

允许访问transcode服务下面的除了logo接口之外的其它接口。

# HTTP接口状态码和错误信息定义

## IDM系统错误码定义

|  |  |  |  |
| --- | --- | --- | --- |
| **服务名** | **错误码** | **错误信息** | **备注** |
| IDM | Success | The request has been successfully processed. | 请求已经被成功处理 |
| MissingParameter | A required parameter for the specified action is not supplied. | 缺少必选参数 |
| InvalidParameter | An invalid or out-of-range value was supplied for the input parameter. | 输入参数不合法 |
| SignatureDoesNotMatch | The request signature we calculated does not match the signature you provided. | 签名不匹配 |
| EntityAlreadyExists | The request was rejected because it attempted to create a resource that already exists. | 实体已存在 |
| NoSuchEntity | The request was rejected because it referenced an entity that does not exist. | 实体不存在 |
| LimitExceeded | The request was rejected because it attempted to create resources beyond the current account limits. | 超出限制 |
| MalformedPolicyDocument | The request was rejected because the policy document was malformed. | Policy格式不正确 |
| Unauthorized | The request access is not authorized. | 请求未被授权 |
| PasswordNotMatch | The provided password does not match the real password. | 密码不匹配 |
| AccessKeyInactive | The request access key is not active. | AccessKey未激活 |
| AccessKeyAlreadyActive | Access key was already active. | AccessKey已经处于激活状态 |
| AccessKeyAlreadyInactive | Access key was already inactive. | AccessKey已经处于冻结状态 |
| AuthFailure | The provided credentials could not be validated. | 鉴权失败 |
| InernalError | The request processing has failed because of an unknown error, exception or failure. | 内部错误 |
| RequestExpired | The request is expired. | 请求已经过期 |
| UnknownError | Unknown error. | 未知错误（所有不在错误码列表中的错误均用此错误码标注） |

## HTTP状态码与IDM错误码对应关系

HTTP状态码与IDM错误码对应关系定义如下：

|  |  |  |
| --- | --- | --- |
| **HTTP状态码** | **含义说明** | **IDM错误码** |
| 200 | OK.  请求已成功，请求所希望的响应头或数据体将随此响应返回。 | Success |
| 400 | Bad Request.  由于包含[语法](http://zh.wikipedia.org/wiki/%E8%AF%AD%E6%B3%95)错误，当前请求无法被服务器理解。除非进行修改，否则客户端不应该重复提交这个请求。 | MissingParameter,InvalidParameter,EntityAlreadyExists,MalformedPolicyDocument |
| 401 | Unauthorized.  当前请求需要用户验证。该响应必须包含一个适用于被请求资源的WWW-Authenticate信息头用以询问用户信息。客户端可以重复提交一个包含恰当的Authorization头信息的请求。如果当前请求已经包含了Authorization证书，那么401响应代表着服务器验证已经拒绝了那些证书。如果401响应包含了与前一个响应相同的身份验证询问，且浏览器已经至少尝试了一次验证，那么浏览器应当向用户展示响应中包含的实体信息，因为这个实体信息中可能包含了相关诊断信息。 |  |
| 403 | Forbidden.  服务器已经理解请求，但是拒绝执行它。与[401响应](http://zh.wikipedia.org/wiki/HTTP%E7%8A%B6%E6%80%81%E7%A0%81#401)不同的是，身份验证并不能提供任何帮助，而且这个请求也不应该被重复提交。如果这不是一个HEAD请求，而且服务器希望能够讲清楚为何请求不能被执行，那么就应该在实体内描述拒绝的原因。当然服务器也可以返回一个[404响应](http://zh.wikipedia.org/wiki/HTTP%E7%8A%B6%E6%80%81%E7%A0%81#404)，假如它不希望让客户端获得任何信息。 | SignatureDoesNotMatch,AccessKeyInactive,AuthFailure,LimitExceeded,RequestExpired,AccessKeyAlreadyActive,AccessKeyAlreadyInactive,PasswordNotMatch,Unauthorized |
| 404 | [Not Found](http://zh.wikipedia.org/wiki/HTTP_404).  请求失败，请求所希望得到的资源未被在服务器上发现。没有信息能够告诉用户这个状况到底是暂时的还是永久的。假如服务器知道情况的话，应当使用[410状态码](http://zh.wikipedia.org/wiki/HTTP%E7%8A%B6%E6%80%81%E7%A0%81#410)来告知旧资源因为某些内部的配置机制问题，已经永久的不可用，而且没有任何可以跳转的地址。404这个状态码被广泛应用于当服务器不想揭示到底为何请求被拒绝或者没有其他适合的响应可用的情况下。 | NoSuchEntity |
| 405 | Method Not Allowed.  请求行中指定的请求方法不能被用于请求相应的资源。该响应必须返回一个Allow头信息用以表示出当前资源能够接受的请求方法的列表。鉴于PUT，DELETE方法会对服务器上的资源进行写操作，因而绝大部分的[网页服务器](http://zh.wikipedia.org/wiki/%E7%BD%91%E9%A1%B5%E6%9C%8D%E5%8A%A1%E5%99%A8)都不支持或者在默认配置下不允许上述请求方法，对于此类请求均会返回405错误。 |  |
| 406 | Not Acceptable.  请求的资源的内容特性无法满足请求头中的条件，因而无法生成响应实体。除非这是一个HEAD请求，否则该响应就应当返回一个包含可以让用户或者浏览器从中选择最合适的实体特性以及地址列表的实体。实体的格式由Content-Type头中定义的媒体类型决定。浏览器可以根据格式及自身能力自行作出最佳选择。但是，规范中并没有定义任何作出此类自动选择的标准。 |  |
| 407 | Proxy Authentication Required.  与[401响应](http://zh.wikipedia.org/wiki/HTTP%E7%8A%B6%E6%80%81%E7%A0%81#401)类似，只不过客户端必须在代理服务器上进行身份验证。代理服务器必须返回一个Proxy-Authenticate用以进行身份询问。客户端可以返回一个Proxy-Authorization信息头用以验证。 |  |
| 408 | Request Timeout.  请求超时。客户端没有在服务器预备等待的时间内完成一个请求的发送。客户端可以随时再次提交这一请求而无需进行任何更改。 |  |
| 409 | Conflict.  由于和被请求的资源的当前状态之间存在冲突，请求无法完成。这个代码只允许用在这样的情况下才能被使用：用户被认为能够解决冲突，并且会重新提交新的请求。该响应应当包含足够的信息以便用户发现冲突的源头。  冲突通常发生于对PUT请求的处理中。例如，在采用版本检查的环境下，某次PUT提交的对特定资源的修改请求所附带的版本信息与之前的某个（第三方）请求向冲突，那么此时服务器就应该返回一个409错误，告知用户请求无法完成。此时，响应实体中很可能会包含两个[冲突版本](http://zh.wikipedia.org/w/index.php?title=%E7%89%88%E6%9C%AC%E5%86%B2%E7%AA%81&action=edit&redlink=1)之间的差异比较，以便用户重新提交[归并](http://zh.wikipedia.org/w/index.php?title=%E5%BD%92%E5%B9%B6&action=edit&redlink=1)以后的新版本。 |  |
| 410 | Gone.  被请求的资源在服务器上已经不再可用，而且没有任何已知的转发地址。这样的状况应当被认为是永久性的。如果可能，拥有链接编辑功能的客户端应当在获得用户许可后删除所有指向这个地址的引用。如果服务器不知道或者无法确定这个状况是否是永久的，那么就应该使用[404状态码](http://zh.wikipedia.org/wiki/HTTP%E7%8A%B6%E6%80%81%E7%A0%81#404)。除非额外说明，否则这个响应是可缓存的。  410响应的目的主要是帮助网站管理员维护网站，通知用户该资源已经不再可用，并且服务器拥有者希望所有指向这个资源的远端连接也被删除。这类事件在限时、增值服务中很普遍。同样，410响应也被用于通知客户端在当前服务器站点上，原本属于某个个人的资源已经不再可用。当然，是否需要把所有永久不可用的资源标记为'410 Gone'，以及是否需要保持此标记多长时间，完全取决于服务器拥有者。 |  |
| 411 | Length Required.  服务器拒绝在没有定义Content-Length头的情况下接受请求。在添加了表明请求消息体长度的有效Content-Length头之后，客户端可以再次提交该请求。 |  |
| 412 | Precondition Failed.  服务器在验证在请求的头字段中给出先决条件时，没能满足其中的一个或多个。这个状态码允许客户端在获取资源时在请求的元信息（请求头字段数据）中设置先决条件，以此避免该请求方法被应用到其希望的内容以外的资源上。 |  |
| 413 | Request Entity Too Large.  服务器拒绝处理当前请求，因为该请求提交的实体数据大小超过了服务器愿意或者能够处理的范围。此种情况下，服务器可以关闭连接以免客户端继续发送此请求。如果这个状况是临时的，服务器应当返回一个Retry-After的响应头，以告知客户端可以在多少时间以后重新尝试。 |  |
| 414 | Request-URI Too Long.  请求的URI长度超过了服务器能够解释的长度，因此服务器拒绝对该请求提供服务。这比较少见，通常的情况包括：   * 本应使用GET方法的表单提交变成了GET方法，导致[查询字符串](http://zh.wikipedia.org/w/index.php?title=%E6%9F%A5%E8%AF%A2%E5%AD%97%E7%AC%A6%E4%B8%B2&action=edit&redlink=1)（[Query String](http://en.wikipedia.org/wiki/Query_String)）过长。 * 重定向URI“黑洞”，例如每次重定向把旧的URI作为新的URI的一部分，导致在若干次重定向后URI超长。 * 客户端正在尝试利用某些服务器中存在的[安全漏洞](http://zh.wikipedia.org/w/index.php?title=%E5%AE%89%E5%85%A8%E6%BC%8F%E6%B4%9E&action=edit&redlink=1)攻击服务器。这类服务器使用固定长度的缓冲读取或操作请求的URI，当GET后的参数超过某个数值后，可能会产生缓冲区溢出，导致任意代码被执行[[1]](http://zh.wikipedia.org/wiki/HTTP%E7%8A%B6%E6%80%81%E7%A0%81#cite_note-1)。没有此类漏洞的服务器，应当返回414状态码。 |  |
| 415 | Unsupported Media Type.  对于当前请求的方法和所请求的资源，请求中提交的实体并不是服务器中所支持的格式，因此请求被拒绝。 |  |
| 416 | Requested Range Not Satisfiable.  如果请求中包含了Range请求头，并且Range中指定的任何数据范围都与当前资源的可用范围不重合，同时请求中又没有定义If-Range请求头，那么服务器就应当返回416状态码。  假如Range使用的是字节范围，那么这种情况就是指请求指定的所有数据范围的首字节位置都超过了当前资源的长度。服务器也应当在返回416状态码的同时，包含一个Content-Range实体头，用以指明当前资源的长度。这个响应也被禁止使用multipart/byteranges作为其Content-Type。 |  |
| 417 | Expectation Failed.  在请求头Expect中指定的预期内容无法被服务器满足，或者这个服务器是一个代理服务器，它有明显的证据证明在当前[路由](http://zh.wikipedia.org/wiki/%E8%B7%AF%E7%94%B1)的下一个节点上，Expect的内容无法被满足。 |  |
| 421 | There are too many connections from your internet address.  从当前客户端所在的[IP地址](http://zh.wikipedia.org/wiki/IP%E5%9C%B0%E5%9D%80)到服务器的连接数超过了服务器许可的最大范围。通常，这里的IP地址指的是从服务器上看到的客户端地址（比如用户的[网关](http://zh.wikipedia.org/wiki/%E7%BD%91%E5%85%B3)或者[代理服务器](http://zh.wikipedia.org/wiki/%E4%BB%A3%E7%90%86%E6%9C%8D%E5%8A%A1%E5%99%A8)地址）。在这种情况下，连接数的计算可能涉及到不止一个[终端](http://zh.wikipedia.org/wiki/%E7%B5%82%E7%AB%AF)用户。 |  |
| 422 | Unprocessable Entity.  请求格式正确，但是由于含有[语义](http://zh.wikipedia.org/wiki/%E8%AF%AD%E4%B9%89)错误，无法响应。 |  |
| 423 | Locked.  当前资源被锁定。 |  |
| 424 | Failed Dependency.  由于之前的某个请求发生的错误，导致当前请求失败，例如PROPPATCH。 |  |
| 426 | Upgrade Required.  客户端应当切换到[TLS/1.0](http://zh.wikipedia.org/wiki/%E4%BC%A0%E8%BE%93%E5%B1%82%E5%AE%89%E5%85%A8)。 |  |
| 500 | Internal Server Error.  服务器遇到了一个未曾预料的状况，导致了它无法完成对请求的处理。一般来说，这个问题都会在服务器的程序码出错时出现。 | InernalError,UnknownError |
| 501 | Not Implemented.  服务器不支持当前请求所需要的某个功能。当服务器无法识别请求的方法，并且无法支持其对任何资源的请求。 |  |
| 502 | Bad Gateway.  作为[网关](http://zh.wikipedia.org/wiki/%E7%BD%91%E5%85%B3)或者[代理](http://zh.wikipedia.org/wiki/%E4%BB%A3%E7%90%86%E6%9C%8D%E5%8A%A1%E5%99%A8)工作的服务器尝试执行请求时，从上游服务器接收到无效的响应。 |  |
| 503 | Service Unavailable.  由于临时的服务器维护或者[过载](http://zh.wikipedia.org/w/index.php?title=%E8%BF%87%E8%BD%BD&action=edit&redlink=1)，服务器当前无法处理请求。这个状况是临时的，并且将在一段时间以后恢复。如果能够预计延迟时间，那么响应中可以包含一个Retry-After头用以标明这个延迟时间。如果没有给出这个Retry-After信息，那么客户端应当以处理[500响应](http://zh.wikipedia.org/w/index.php?title=500_Internal_Error&action=edit&redlink=1)的方式处理它。 |  |

## 错误信息格式定义

* **消息格式**

Json格式。

字段定义如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段类型** | **字段含义** | **是否必选** | **备注** |
| errorCode | String | 错误码 | 必选 | 定义参见《IDM错误码设计文档》。 |
| errorMessage | String | 错误信息 | 必选 | 定义参见《IDM错误码设计文档》。 |

* **消息样例**

{"errorCode":"MissingParameter","errorMessage":"A required parameter for the specified action is not supplied."}