

Group

List & Create

HTTP REQUEST: GET

Gets all Groups in the Organization identified by orgID UUID.

```
GET /api/1.0/org/{orgId}/groups
```

URI PARAMETERS

Name	Type	Description
orgId	string	The Organization ID (UUID) for which you are querying.

RESPONSES

Code	Description
200	Success.
401	This error will occur if calling user does not have access right for the API.
404	This error will occur if the org identified by orgId does not exist.
500	A server error occurred. Check the logs for more details.

200 Reponse Body

```

{
  "status": {
    "i18n_message": "response.ok",
    "message": "OK"
  },
  "response": [
    {
      "ID": "306a42c9-a7f3-48c3-743c-10015e29a672",
      "OrgID": "default",
      "Name": "My Group",
      "NumberOfUsers": 0
    },
    {
      "ID": "106b52fb-43e3-4ffb-491a-bb6497300c54",
      "OrgID": "default",
      "Name": "My Other Group",
      "NumberOfUsers": 0
    }
  ]
}

```

HTTP REQUEST: POST

Create a new Group in the Organization

POST /api/1.0/org/{orgId}/groups

URI PARAMETERS

Name	Type	Description
orgId	string	The Organization ID (UUID) for which you are querying.

RESPONSES

Code	Description
200	Success.
401	This error will occur if calling user does not have access right for the API.

404	This error will occur if the org identified by orgId does not exist.
500	A server error occurred. Check the logs for more details.

Request Body

```
{
  "name": "My Third Group"
}
```

200 Reponse Body

```
{
  "status": {
    "i18n_message": "response.ok",
    "message": "OK"
  },
  "response": {
    "ID": "2f81a670-6c9b-4af4-71fd-7a79adda0da2",
    "OrgID": "default",
    "Name": "My Third Group"
  }
}
```

Update & Delete

HTTP REQUEST: POST

Update a Group identified by groupId.

```
POST /api/1.0/org/{orgId}/groups/{groupId}
```

URI PARAMETERS

Name	Type	Description
orgId	string	The Organization ID (UUID) for which you are querying.
groupId	string	The Group ID (UUID) for which you are querying.

RESPONSES

Code	Description
200	Success.
401	This error will occur if calling user does not have access right for the API.
404	This error will occur if the org identified by orgId does not exist.
500	A server error occurred. Check the logs for more details.

Request Body

```
{
  "name": "My New Group Name"
}
```

200 Reponse Body

```
{
  "status": {
    "i18n_message": "response.ok",
    "message": "OK"
  },
  "response": {
    "ID": "2f81a670-6c9b-4af4-71fd-7a79adda0da2",
    "OrgID": "default",
    "Name": "My New Group Name"
  }
}
```

HTTP REQUEST: DELETE

Deletes the Group identified by groupId.

```
DELETE /api/1.0/org/{orgId}/groups/{groupId}
```

URI PARAMETERS

Name	Type	Description
------	------	-------------

orgId	string	The Organization ID (UUID) for which you are querying.
groupId	string	The Group ID (UUID) for which you are querying.

RESPONSES

Code	Description
200	Success.
401	This error will occur if calling user does not have access right for the API.
404	This error will occur if the org identified by orgId does not exist.
500	A server error occurred. Check the logs for more details.

200 Reponse Body

```
{
  "status": {
    "i18n_message": "response.ok",
    "message": "OK"
  },
  "response": {
    "ID": "2f81a670-6c9b-4af4-71fd-7a79adda0da2",
    "OrgID": "default",
    "Name": "My New Group Name"
  }
}
```

User Management

HTTP REQUEST: GET

Get the list of users that exist in this group

```
GET /api/1.0/org/{orgId}/groups/{groupId}/users
```

URI PARAMETERS

Name	Type	Description
orgId	string	The Organization ID (UUID) for which you are querying.
groupId	string	The Group ID (UUID) for which you are querying.

RESPONSES

Code	Description
200	Success.
401	This error will occur if calling user does not have access right for the API.
404	This error will occur if the org identified by orgId does not exist.
500	A server error occurred. Check the logs for more details.

200 Reponse Body

```
{
  "status": {
    "i18n_message": "response.ok",
    "message": "OK"
  },
  "response": {
    "users": [
      {
        "user_id": "e381667f-8049-46f6-4916-e5b8a93ffe87",
        "name": "Kristi Long",
        "email": "kristi@atscale.com",
        "auth_username": "kristi@atscale.com",
        "super_user": true,
        "api_super_user": false,
        "session_password": ""
      },
      {
        "user_id": "27354c24-f5b8-4fbb-6e82-58a8b67b12c5",
        "name": "Mike Aldrich",
```

```

    "email": "mike@atscale.com",
    "auth_username": "mike@atscale.com",
    "super_user": false,
    "api_super_user": false,
    "session_password": ""
  },
  {
    "user_id": "49d46e42-ced3-4fe1-41da-62a486791416",
    "name": "Rafael Lopez",
    "email": "rafael@atscale.com",
    "auth_username": "rafael@atscale.com",
    "super_user": true,
    "api_super_user": false,
    "session_password": ""
  },
  {
    "user_id": "c8aec429-0218-45af-5704-413406f43232",
    "name": "Shanti Chand",
    "email": "shanti@atscale.com",
    "auth_username": "shanti@atscale.com",
    "super_user": true,
    "api_super_user": false,
    "session_password": ""
  }
]
}
}

```

HTTP REQUEST: POST

Sets Users in Group to the list of UUID's.

POST /api/1.0/org/{orgId}/groups/{groupId}/users

URI PARAMETERS

Name	Type	Description
orgId	string	The Organization ID (UUID) for which you are querying.
groupId	string	The Group ID (UUID) for which you are querying.

RESPONSES

Code	Description
200	Success.
401	This error will occur if calling user does not have access right for the API.
404	This error will occur if the org identified by orgId does not exist.
500	A server error occurred. Check the logs for more details.

200 Reponse Body

```
{
  "status": {
    "i18n_message": "response.ok",
    "message": "OK"
  },
  "response": {
    "ID": "2f81a670-6c9b-4af4-71fd-7a79adda0da2",
    "OrgID": "default",
    "Name": "My New Group Name"
  }
}
```