Fetching organisations via the API

The API provides methods to query the organisational structure for a domain. This allows a client to obtain information about a given organisation, and also perform operations on accounts that belong to the organisation.

Fetching an organisation

Prerequisites

To fetch an organisation, a client application must

- authenticate to the API as described in Authenticating to the API.
- obtain a link to an organisation by following a link from a domain, an account object, or by querying an organisation as outlined below.

Procedure

To view information about an organisation, perform a GET request to an organisation resource. To find the resource for the organisation to which the current session belongs, follow the link with a relation of ‘organisation:root’ from the entry-point resource. This will be a URL of the form:

> /api/v1/example.org/organisation/<id>

This request returns an application/vnd.eduserv.iam.admin.organisation-v1+json object.

Example

```
Request:
GET /api/v1/example.org/organisation/12345 HTTP/1.1
Authorization: OAApiKey <api-key>
Response (success):
HTTP/1.1 200 OK
Content-Type: application/vnd.eduserv.iam.admin.organisation-v1+json

{
    "id" : "1234",
    "name" : "org 1",
    "ipRanges" : [ "192.168.1.*" ],
    "attributes" : { "alternativeNames" : [ "Alternative name" ],
                     "emailDomains" : [ "example.com", "example.org" ],
                     "ipRanges" : [ "*.example.com", "*.example.org" ]
    },
    "links" : [
               { "rel" : "self",
                 "type" : "application/vnd.eduserv.iam.admin.organisation-v1+json",
                 "href" : "/api/v1/example.org/organisation/1234",
                 "method" : "get"
               },
               { "rel" : "up",
                 "type" : "application/vnd.eduserv.iam.admin.organisation-v1+json",
                 "href" : "/api/v1/example.org/organisation/0000",
                 "method" : "get"
               },
               { "rel" : "down",
                 "type" : "application/vnd.eduserv.iam.admin.groupList-v1+json",
                 "href" : "/api/v1/example.org/organisation/4567/groups",
                 "method" : "get",
               },
               { "rel" : "organisation:query",
                 "type" : "application/vnd.eduserv.iam.admin.organisationList-v1+json",
                 "href" : "/api/v1/example.org/organisation/4567/query",
                 "method" : "get"
               },
               { "rel" : "add",
                 "type" : "application/vnd.eduserv.iam.accountRequest-v1+json",
                 "href" : "/api/v1/example.org/organisation/4567/accounts/create/personal",
                 "method" : "post"
               }
    ]
}
```

Response codes

HTTP Response Code	Description
200	The request was successful.
400	The request was invalid.
404	The organisation does not exist.

application/vnd.eduserv.iam.admin.organisation-v1+json object

Object field	Description
name	The display name of the organisation.
ipRanges	A list of IP address ranges from which administration of the organisation is restricted.
attributes	A list of attributes relating to the organisation. The attributes returned are defined in the organisation schema.

Querying organisations

Within the OpenAthens system, organisations are arranged in a hierarchy, as outlined in section 3.4. The API provides a means to query this hierarchical structure by providing access to a list of sub-organisations beneath a known parent organisation.

List sub-organisations

Prerequisites

To query for organisations, a client application must

- authenticate to the API as described in Authenticating to the API.
- obtain a parent organisation object, as described above.

Procedure

To query for sub-organisations under the control of a particular parent organisation, follow the link with a relation of 'organisation:query' from the entry-point resource, or from a parent organisation object. Perform a GET request to:

/api/v1/example.org/organisation/<id>/query

The depth of the query can be controlled via the querystring (see below). The returned data is a flat (non-hierarchical) list of sub-organisations. This can be used by a self-registration interface to cache a type-ahead search for organisation names, without concern for the account hierarchy.

Request querystring

The request can be modified by adding the following querystring parameters to the URL:

Querystring Parameter	Description
attributes	A list of *additional* attributes to return as part of the result. This avoids a client having to make many additional requests to query attributes for an individual organisation if additional metadata is required at the time the list is generated. If more than one attribute is required, multiple parameters may be added (see example, below).
depth	The depth of sub-organisations to search. A depth of 1 (the default) will list organisations directly under the organization being queried. A depth of -1 will search all levels.
filter	An additional filter to apply to results. The result-list will only contain organisations where the organisation name **or** the values of any requested attributes *contains* the filter value supplied by this parameter. The filter is matched case-insensitively.
includeAll	By default only organisations with a unique public identifier assigned to them will be returned by the query. Setting this to 'true' will include all organisations, whether public or not. Defaults to 'false'.

Example

```
Request:
GET /api/v1/example.org/organisation/12345/query?attributes=alternativeName&attributes=emailDomain&depth=1 HTTP
/1.1
Authorization: OAApiKey <api-key>
Response (success):
HTTP/1.1 200 OK
Content-Type: application/vnd.eduserv.iam.admin.organisationList-v1+json

{
  "organisations": [
                     { "id" : "4567",
                       "href" : "/api/v1/example.org/organisation/4567",
                       "name" : "org 1",
                       "attributes": {
                         "alternativeNames" : [ "blah" ]
                       }
                     },
                     { "id" : "7654",
                       "href" : "/api/v1/example.org/organisation/7654",
                       "name" : "org 2"
                     }
  ]
}
```

List available permission sets for an organisation

Prerequisites

To list the permission sets belonging to an organisation, a client application must

- authenticate to the API as described in Authenticating to the API.
- obtain a parent organisation object, as described above.

Procedure

To list the permission sets of a particular parent organisation, follow the link with a relation of ‘organisation:permission-sets’ from the entry-point resource, or from a parent organisation object.

Perform a GET request to:

> /api/v1/example.org/organisation/<id>/permission-sets

Request querystring

The request can be modified by adding the following querystring parameter to the URL:

Querystring Parameter	Description
includeCounts	Setting this to ‘true’ will fetch the number of allocated users and number of allocated resources for each permission set. Defaults to ‘false’.

Example

Error rendering macro 'code': Invalid value specified for parameter 'com.atlassian.confluence.ext.code.render.InvalidValueException'

```
Request:
GET /api/v1/example.org/organisation/12345/permission-sets?includeCounts=true HTTP/1.1
Authorization: OAApiKey <api-key>
Response (success):
HTTP/1.1 200 OK
Content-Type: application/vnd.eduserv.iam.admin.admin.permissionSetList-v1+json

{
"total": 2,
    "number": 0,
    "offset": 0,
    "permissionSets": [
        {
            "id": "12345",
            "name": "ath#pset001",
            "description": "Default permission set",
            "attributes": {},
            "created": "2015-03-03T12:49:00Z",
            "modified": "2015-08-19T10:38:00Z",
            "numberOfAllocatedUsers": 32,
            "numberOfAllocatedResources": 15,
            "default": true
        },
        {
            "id": "24680",
            "name": "ath#pset002",
            "description": "Another permission set",
            "attributes": {},
            "created": "2011-10-19T13:40:00Z",
            "modified": "2015-08-04T16:00:00Z",
            "numberOfAllocatedUsers": 5,
            "numberOfAllocatedResources": 2,
            "default": false
        }

}
```

See also:

- API overview
- Authenticating to the API
- API entry-point
- Fetching attribute schemas via the API
- Fetching organisations via the API
- Fetching Groups via the API
- Account management via the API
- API bulk operations
- Fetching available service providers via the API
- Generating authentication tokens for end-users via the API
- API usage examples