

Get Members

Wiki Home	Grouper Release Announcements	Grouper Guides	Grouper Deployment Guide	Community Contributions	Internal Developer Resources
---------------------------	---	--------------------------------	--	---	--

[Grouper Web Services](#)

Description

Get members will retrieve subjects assigned to a group.

Features

- Can base member list based on memberfilter (e.g. All, Immediate, Effective)
- Lookup subjects by subject lookup (by id, source, identifier, etc)
- Lookup groups by group lookup (by name or uuid)
- Returns group / subject information, can be detailed or not
- Can actAs another user
- For 2.0+, you can pass in pointInTimeFrom and pointInTimeTo to get the member list at a certain point in time in the past, or in a date range. This should be formatted: yyyy/MM/dd HH:mm:ss.SSS
- In 2.1.3+ you can [page](#) or [sort the results](#)

Get members Lite service

- Accepts one group to get members for
- Documentation: [SOAP](#) (click on getMembersLite), [REST](#) (click on getMembersLite)
- For REST, the request can put data in query string (in URL or request body)
- REST request (colon is escaped to %3A): GET /grouper-ws/servicesRest/v1_3_000/groups/aStem%3AaGroup/members
 - Note: if passing data in request body e.g. actAs, use a POST
- (see documentation above for details): [Request object](#), [response object](#)
- [Response codes](#)
- [Samples](#) (all files with "Lite" in them, click on "download" to see file)

Get members service

- Accepts multiple groups to retrieve lists of lists of subjects
- Documentation: [SOAP](#) (click on getMembers), [REST](#) (click on getMembers)
- REST request (colon is escaped to %3A): POST /grouper-ws/servicesRest/v1_3_000/groups/aStem%3AaGroup
- (see documentation above for details): [Request object](#), [response object](#)
- [Response codes overall](#), [response codes for each assignment](#)
- Returns an overall status, and a status for each assignment
- [Samples](#) (all files without "Lite" in them, click on "download" to see files)

Example get netIds from subject source

- Get members from a Group in json for a certain source and include netId in response
- https://grouperws.school.edu/grouper-ws/servicesRest/v2_6_000/groups/penn%3Aisc%3Aait%3Aapps%3Aopenshift%3Aservice%3AanonprodPolicy%3Aroot%3Aapp%3Aacim-bridge%3AXCat2%3Ais_student/members?wsLiteObjectType=WsRestGetMembersLiteRequest&sourceIds=pennperson&subjectAttributeNames=pennname
- Response

```
{
  "WsGetMembersLiteResult":{
    "resultMetadata":{
      "success":"T",
      "resultCode":"SUCCESS",
      "resultMessage":"Success for: clientVersion: 2.6.0, wsGroupLookups: Array size: 1: [0]:
WsGroupLookup[pitGroups=[],groupName=penn:isc:ait:apps:openshift:service:nonprodPolicy:root:app:cim-
bridge:XCat2:is_student]\n\n, memberFilter: All, includeSubjectDetail: false, actAsSubject: null,
fieldName: null, subjectAttributeNames: Array size: 1: [0]: pennname\n\n, paramNames: \n, params:
null\n, sourceIds: Array size: 1: [0]: pennperson\n\n, pointInTimeFrom: null, pointInTimeTo: null,
pageSize: null, pageNumber: null, sortString: null, ascending: null"
    },
    "wsGroup":{
      "extension":"is_student",
      "displayName":"penn:isc:ait:apps:openshift:service:nonprodPolicy:root:app:cim-bridge:XCat2:
is_student",
      "uuid":"82799bf5a1694d9c811e53b2c9ff52ed",
      "enabled":"T",
      "displayExtension":"is_student",
    }
  }
}
```

```

        "name":"penn:isc:ait:apps:openshift:service:nonprodPolicy:root:app:cim-bridge:XCat2:is_student",
        "typeOfGroup":"group",
        "idIndex":"559379"
    },
    "subjectAttributeNames":[
        "pennname"
    ],
    "responseMetadata":{
        "serverVersion":"2.5.55",
        "resultWarnings":", Client version: 2.6.0 is greater than (major/minor) server version: 2.5.55",
        "millis":"69"
    },
    "wsSubjects":[
        {
            "sourceId":"pennperson",
            "success":"T",
            "attributeValues":[
                "wabe"
            ],
            "name":"Terry Johnson",
            "resultCode":"SUCCESS",
            "id":"212346",
            "memberId":"bf1e6c3b068147c9a7c90c252d4cf161"
        },
        {
            "sourceId":"pennperson",
            "success":"T",
            "attributeValues":[
                "cabd"
            ],
            "name":"Daniel J Johnson",
            "resultCode":"SUCCESS",
            "id":"312347",
            "memberId":"206934ec147e4a4eb73f3367e1dba3f9"
        },
        {
            "sourceId":"pennperson",
            "success":"T",
            "attributeValues":[
                "mabs"
            ],
            "name":"Anome Johnson",
            "resultCode":"SUCCESS",
            "id":"312347",
            "memberId":"9ec3b16b-03d3-4742-be63-bbb9c2526631"
        },
        {
            "sourceId":"pennperson",
            "success":"T",
            "attributeValues":[
                "wabh"
            ],
            "name":"William G Johnson",
            "resultCode":"SUCCESS",
            "id":"612341",
            "memberId":"096da10b-0fc8-4a51-bfbc-deef91707ac6"
        }
    ]
}
}
}

```