

Call REST services from CLI

1. [Apache Syncope >= 2.0.X](#)
2. [Apache Syncope 1.2.X](#)
 1. [Create user](#)
 2. [Update user](#)
 3. [Delete user](#)
 4. [Export database content](#)
 5. [Change configuration parameter value](#)
3. [Apache Syncope <= 1.1.X](#)
 1. [Create user](#)
 2. [Export database content](#)
 3. [Change configuration parameter value](#)

In the following it is assumed that:

- [curl](#) is the reference tool for CLI invocation
- `admin:password` are the credentials for accessing Syncope core
- Syncope core runs on <http://localhost:9080/syncope>

Apache Syncope >= 2.0.X

Details are available in the [Reference Guide](#).



With Apache Syncope 2.0.0, besides using [curl](#) as with earlier versions, a brand new [CLI](#) tool is available.



For more details please refer to the [REST API reference](#), which is also available at <http://localhost:9080/syncope/>.



If selected for deployment, the [Swagger](#) extension is also available at <http://localhost:9080/syncope/swagger/>.

Apache Syncope 1.2.X



For more details please refer to the [REST API reference](#), which is also available at <http://localhost:9080/syncope/rest/doc/>.

Create user

```
curl -X POST -u admin:password -H "Accept: application/json" -H "Content-Type: application/json" \
--data @/tmp/newuser.json http://localhost:9080/syncope/rest/users
```

where `/tmp/newuser.json` looks like:

```
{
  "id": 0,
  "password": "password123",
  "username": "myusername@apache.org",
  "securityQuestion": null,
  "securityAnswer": null,
  "attributes": [
    {
      "schema": "fullname",
      "readonly": false,
      "values": [
        "myusername@apache.org"
      ]
    },
    {
      "schema": "loginDate",
      "readonly": false,
      "values": [
        "2015-11-12"
      ]
    }
  ]
}
```

To link the user to an existing resource, include this in the root json hash:

```
"resources": [
  "ws-target-resource-1"
]
```

You might want to add

```
-H "Prefer: return-no-content"
```

to avoid Syncope core populating the HTTP response with the created user entry.

Update user

```
curl -X POST -u admin:password -H "Accept: application/json" -H "Content-Type: application/json" \
--data @/tmp/updateuser.json http://localhost:9080/syncope/rest/users/100
```

where 100 is the id of the user to be deleted and /tmp/updateuser.json looks like:

```
{
  "id": 100,
  "username": "newUsername",
  "password": "updatedPassword",
  "attributesToRemove": [
    "surname"
  ],
  "attributesToUpdate": [
    {
      "schema": "surname",
      "valuesToBeAdded": [
        "New surname"
      ],
      "valuesToBeRemoved": []
    }
  ]
}
```

You might want to add

```
-H "Prefer: return-no-content"
```

to avoid Syncope core populating the HTTP response with the updated user entry.

Delete user

```
curl -X DELETE -u admin:password -H "Accept: application/json" http://localhost:9080/syncope/rest/users/100
```

where 100 is the id of the user to be deleted.

You might want to add

```
-H "Prefer: return-no-content"
```

to avoid Syncope core populating the HTTP response with the deleted user entry.

Export database content

```
curl -X GET -u admin:password -o content.xml http://localhost:9080/syncope/rest/configurations/stream
```

Change configuration parameter value

```
curl -X PUT -u admin:password -H "Accept: application/json" -H "Content-Type: application/json" \
-d '{"schema": "selfRegistration.allowed", "values": ["true"]}' http://localhost:9080/syncope/rest/configurations
/selfRegistration.allowed
```

Apache Syncope <= 1.1.X

Create user

```
curl -u admin:password -H "Content-Type: application/json" --data @/tmp/newuser.json http://localhost:9080
/syncope/rest/user/create.json
```

where /tmp/newuser.json looks like:

```
{
  "id": "0",
  "username": "myusername",
  "password": "test123",
  "attributes": [
    {
      "schema": "userId",
      "values": [
        "example1@example.com"
      ]
    },
    {
      "schema": "surname",
      "values": [
        "My Surname"
      ]
    }
  ]
}
```

To link the user to an existing resource, include this in the root json hash:

```
"resources": [  
  "ws-target-resource-1"  
]
```

Export database content

```
curl -u admin:password -o content.xml -H "Content-Type: application/xml" -X GET http://localhost:9080/syncope/cxf/configurations/stream
```

Change configuration parameter value

```
curl -u admin:password -H "Content-Type: application/json" \  
-d '{"key":"smtp.host","value":"smtp.ngi.it"}' http://localhost:9080/syncope/rest/configuration/update.json
```