Security API

When secure mode is enabled in the Staging Repository, the security API is available to restrict access per storage unit. A single *master* administration user is designated in the configuration file of the staging repository. This user will be able to add other administration users or regular users that can have read and/or write access to specific storage units.

On this page:

Secure Configuration

To enable secure mode and the security API, the Staging Repository requires the **secure** node in the configuration file to be defined. **secure** node will contain the **adminUser** which will be the first user to have admin access to the Staging Repository. The value of **adminUser** is the CN (common name) in the client certificate that the admin user will use to authenticate against the Staging Repository. Besides the adminUser, the secure node defines the server certificate and CA certificate to run the Staging Repository with SSL enabled.

```
{
        ...,
        secure: {
                  adminUser: 'HorizonAspire',
                keyLocation: 'config/sslcerts/sr/sr_server_key.pem',
                certLocation: 'config/sslcerts/sr/sr_server_cert.crt',
                caLocation: 'config/sslcerts/sr/cacert.crt',
                passphrase: 'mypass123',
                requestCert: true,
                rejectUnauthorized: false
        },
        ...
}
```

When the **adminUser** or any other administration user is adding a new user, the username that is used corresponds to the CN (common name) of the client certificate of the users to add.

Add User Permissions to Storage Unit

Assigns a user with read and/or write rights to a specific storage unit.

NOTE: An administration user will always be able to access/read/write to storage units via the transaction api.

Request

The add user permissions to storage unit PUT/GET request requires the rights to assign (r, w, rw), the username (CN of the client certificate) and the storage unit to assign the user permissions to.

```
PUT security/add/<permissions>/<user>/<storageUnit>
```

Response

Returns a 200 response code and an OK response message if the user permissions were correctly assigned to the storage unit.

```
{"message": "OK"}
```

Remove User Permissions from Storage Unit

Removes user permissions from a specific storage unit.

NOTE: An administration user will always be able to access/read/write to storage units via the transaction api.

Request

The remove user permissions from storage unit DELETE/GET request requires the rights to remove (r, w, rw), the username (CN of the client certificate) and the storage unit to unassign the user permissions from.

```
DELETE security/remove/<permissions>/<user>/<storageUnit>
```

Response

Returns a 200 response code and an OK response message if the user permissions were correctly unassigned from the storage unit.

```
{"message": "OK"}
```

Add Administration User

Adds a user as an administrator user.

NOTE: An administration user will always be able to access/read/write to storage units via the transaction api.

Request

The add administration user PUT/GET request requires the username (CN of the client certificate) to be added as administrator.

```
PUT security/addAdmin/<user>
```

Response

Returns a 200 response code and an OK response message if the user was correctly added as an administrator.

```
{"message": "OK"}
```

Remove Administration User

Removes a user as an administrator user.

Request

The remove administration user DELETE/GET request requires the username (CN of the client certificate) to be removed as administrator.

```
DELETE security/removeAdmin/<user>
```

Response

Returns a 200 response code and an OK response message if the user was correctly removed as an administrator.

```
{"message": "OK"}
```