Arrowhead Authorization Core Service 4.1.3 Release Notes

In Arrowhead version 4.1.3, the whole codebase of the Arrowhead framework has been rewritten using Spring Boot technology. We believe the new version is better both in code quality, performance and robustness. It is easier to use, maintain and improve.
However, this means that Arrowhead version 4.1.3 is NOT compatible with providers and consumers written for version 4.1.2 because the changes affect not just the backend but the public interface of the core services, too.
This document describes the new public API of the Arrowhead Authorization Core Service 4.1.3 and shows the key differences between the new interface and the old one.

Client Services

These services can be used by providers and consumers of the Arrowhead Cloud.

Uri: /authorization/echo
Type: GET

Returns a “Got it” message with the purpose of testing the core service availability.

4.1.2 version: GET /authorization/mgmt
It was only available for system operator of the local cloud.

Uri: /authorization/publickey
Type: GET

Returns the public key of the Authorization core service as a (Base64 encoded) text. This service is necessary for providers if they want to utilize the token based security.

4.1.2 version: GET /authorization/mgmt/publickey
It was only available for system operator of the local cloud.

Management Services

These services can only be used by the system operator of the local cloud. All date fields contain the text representation of a UTC timestamp.

Uri: /authorization/mgmt/intracloud
Type: GET
Query params:

- *page* – zero-based page index (optional),
- *item_per_page* – maximum number of items returned (optional),
- *sort_field* – sort field (optional, default: id, possible values: id, createdAt, updatedAt),
- *direction* – direction of sorting (optional, default: ASC, possible values: ASC or DESC)

Returns a page of intra-cloud related authorization records. If page and item_per_page are not defined, returns all records.

Returned JSON structure:

```
{
  "count": 0,
  "data": [
    {
      "id": 0,
      "consumerSystem": {
        "id": 0,
        "systemName": "string",
        "address": "string",
        "port": 0,
        "authenticationInfo": "string",
        "createdAt": "string",
        "systemName": "string",
        "updatedAt": "string"
      },
      "providerSystem": {
        "id": 0,
        "systemName": "string",
        "address": "string",
        "port": 0,
        "authenticationInfo": "string",
        "createdAt": "string",
        "updatedAt": "string"
      },
      "serviceDefinition": {
        "id": 0,
        "serviceDefinition": "string",
        "createdAt": "string",
        "updatedAt": "string"
      },
      "interfaces": [
        {
          "id": 0,
          "interfaceName": "string",
          "createdAt": "string",
          "updatedAt": "string"
        }
      ],
      "createdAt": "string",
      "updatedAt": "string"
    }
  ]
}
```

Authorization is a little stricter than before: the access now depends on specific interfaces besides provider and service.

4.1.2 version: GET /authorization/mgmt/intracloud
This version always returned all records in an array of JSON objects. The objects did not contain any time information. Access didn't depend on interface.

Uri: /authorization/mgmt/intracloud
Type: POST

Creates intra-cloud authorization records and returns the newly created records.

Input JSON structure:

```
{
  "consumerId": 0,
  "providerIds": [
    0
  ],
  "interfaceIds": [
    0
  ],
  "serviceDefinitionIds": [
    0
  ]
}
```

This is a very general stucture, however only two possible combinations are allowed:

- one provider id, one interface id with multiple service definition ids, and
- multiple provider ids, multiple interface ids with one service definition id.

Returned JSON structure:

```
{
  "count": 0,
  "data": [
    {
      "id": 0,
      "consumerSystem": {
        "id": 0,
        "systemName": "string",
        "address": "string",
        "port": 0,
        "authenticationInfo": "string",
        "createdAt": "string",
        "systemName": "string",
        "updatedAt": "string"
      },
      "providerSystem": {
        "id": 0,
        "systemName": "string",
        "address": "string",
        "port": 0,
        "authenticationInfo": "string",
        "createdAt": "string",
        "updatedAt": "string"
      },
      "serviceDefinition": {
        "id": 0,
        "serviceDefinition": "string",
        "createdAt": "string",
        "updatedAt": "string"
      },
      "interfaces": [
        {
          "id": 0,
          "interfaceName": "string",
          "createdAt": "string",
          "updatedAt": "string"
        }
      ],
      "createdAt": "string",
      "updatedAt": "string"
    }
  ]
}
```

4.1.2 version: POST /authorization/mgmt/intracloud
This version required whole JSON objects as consumer, provider and service instead of ids and didn't use interface restrictions.

Uri: /authorization/mgmt/intracloud/{id}
Type: GET

Returns the intra-cloud related authorization record specified by the id path parameter.

Returned JSON structure:

```
{
  "id": 0,
  "consumerSystem": {
    "id": 0,
    "systemName": "string",
    "address": "string",
    "port": 0,
    "authenticationInfo": "string",
    "createdAt": "string",
    "systemName": "string",
    "updatedAt": "string"
  },
  "providerSystem": {
    "id": 0,
    "systemName": "string",
    "address": "string",
    "port": 0,
    "authenticationInfo": "string",
    "createdAt": "string",
    "updatedAt": "string"
  },
  "serviceDefinition": {
    "id": 0,
    "serviceDefinition": "string",
    "createdAt": "string",
    "updatedAt": "string"
  },
  "interfaces": [
    {
      "id": 0,
      "interfaceName": "string",
      "createdAt": "string",
      "updatedAt": "string"
    }
  ],
  "createdAt": "string",
  "updatedAt": "string"
}
```

4.1.2 version: GET /authorization/mgmt/intracloud/{id}
The returned structure did not contain time information and interface restrictions.

Uri: /authorization/mgmt/intracloud/{id}
Type: DELETE

Removes the intra-cloud related authorization record specified by the id path parameter.

4.1.2 version: DELETE /authorization/mgmt/intracloud/{id}
Basically was the same than the new version.

Uri: /authorization/mgmt/intercloud
Type: GET
Query params:

- *page* – zero-based page index (optional),
- *item_per_page* – maximum number of items returned (optional),
- *sort_field* – sort field (optional, default: id, possible values: id, createdAt, updatedAt),
- *direction* – direction of sorting (optional, default: ASC, possible values: ASC or DESC)

Returns a page of inter-cloud related authorization records. If page and item_per_page are not defined, returns all records.
Returned JSON structure:

```
{
  "count": 0,
  "data": [
    {
      "id": 0,
      "cloud": {
        "id": 0,
        "operator": "string",
        "name": "string",
        "authenticationInfo": "string",
        "secure": true,
        "neighbor": true,
        "ownCloud": true,
        "createdAt": "string",
        "updatedAt": "string"
      },
      "provider": {
        "id": 0,
        "systemName": "string",
        "address": "string",
        "port": 0,
        "authenticationInfo": "string",
        "createdAt": "string",
        "updatedAt": "string"
      },
      "serviceDefinition": {
        "id": 0,
        "serviceDefinition": "string",
        "createdAt": "string",
        "updatedAt": "string"
      },
      "interfaces": [
        {
          "id": 0,
          "interfaceName": "string",
          "createdAt": "string",
          "updatedAt": "string"
        }
      ],
      "createdAt": "string",
      "updatedAt": "string"
    }
  ]
}
```

Authorization is stricter than before: the access now depends on specific provider and interfaces besides service.

4.1.2 version: GET /authorization/mgmt/intercloud
This version always returned all records in an array of JSON objects. The objects did not contain any time information. Access didn't depend on provider and interface.

Uri: /authorization/mgmt/intercloud
Type: POST

Creates inter-cloud authorization records and returns the newly created records.

Input JSON structure:

```
{
  "cloudId": 0,
  "providerIdList": [
    0
  ],
  "interfaceIdList": [
    0
  ],
  "serviceDefinitionIdList": [
    0
  ]
}
```

This is a very general stucture, however only two possible combinations are allowed:
one provider id, one interface id with multiple service definition ids, and
multiple provider ids, multiple interface ids with one service definition id.

Returned JSON structure:

```
{
  "count": 0,
  "data": [
    {
      "id": 0,
      "cloud": {
        "id": 0,
        "operator": "string",
        "name": "string",
        "authenticationInfo": "string",
        "secure": true,
        "neighbor": true,
        "ownCloud": true,
        "createdAt": "string",
        "updatedAt": "string"
      },
      "provider": {
        "id": 0,
        "systemName": "string",
        "address": "string",
        "port": 0,
        "authenticationInfo": "string",
        "createdAt": "string",
        "updatedAt": "string"
      },
      "serviceDefinition": {
        "id": 0,
        "serviceDefinition": "string",
        "createdAt": "string",
        "updatedAt": "string"
      },
      "interfaces": [
        {
          "id": 0,
          "interfaceName": "string",
          "createdAt": "string",
          "updatedAt": "string"
        }
      ],
      "createdAt": "string",
      "updatedAt": "string"
    }
  ]
}
```

4.1.2 version: POST /authorization/mgmt/intercloud
This version required whole JSON objects as consumer cloud and service instead of ids and didn't use provider and interface restrictions.

Uri: /authorization/mgmt/intercloud/{id}
Type: GET

Returns the inter-cloud related authorization record specified by the id path parameter.

Returned JSON structure:

```
{
  "id": 0,
  "cloud": {
    "id": 0,
    "operator": "string",
    "name": "string",
    "authenticationInfo": "string",
    "secure": true,
    "neighbor": true,
    "ownCloud": true,
    "createdAt": "string",
    "updatedAt": "string"
  },
  "provider": {
    "id": 0,
    "systemName": "string",
    "address": "string",
    "port": 0,
    "authenticationInfo": "string",
    "createdAt": "string",
    "updatedAt": "string"
  },
  "serviceDefinition": {
    "id": 0,
    "serviceDefinition": "string",
    "createdAt": "string",
    "updatedAt": "string"
  },
  "interfaces": [
    {
      "id": 0,
      "interfaceName": "string",
      "createdAt": "string",
      "updatedAt": "string"
    }
  ],
  "createdAt": "string",
  "updatedAt": "string"
}
```

4.1.2 version: GET /authorization/mgmt/intercloud/{id}
The returned structure did not contain time information, provider and interface restrictions.

Uri: /authorization/mgmt/intercloud/{id}
Type: DELETE

Removes the inter-cloud related authorization record specified by the id path parameter.

4.1.2 version: DELETE /authorization/mgmt/intercloud/{id}
Basically was the same than the new version.

The following services are no longer exist:

- GET /authorization/mgmt/intracloud/systemId/{systemId}/services
- GET /authorization/mgmt/intracloud/systemId/{systemId}
- GET /authorization/mgmt/intracloud/servicedef/{serviceDefinition}
- PUT /authorization/mgmt/intracloud
- DELETE /authorization/mgmt/intracloud/systemId/{systemId}
- GET /authorization/mgmt/intercloud/operator/{operator}/cloudname/{cloudName}/services
- GET /authorization/mgmt/intercloud/operator/{operator}/cloudname/{cloudName}
- GET /authorization/mgmt/intercloud/servicedef/{serviceDefinition}
- PUT /authorization/mgmt/intercloud
- DELETE /authorization/mgmt/intercloud/operator/{operator}/cloudname/{cloudName}

Private Services

These services can only be used by other core services, therefore they are not part of the public API.