Preface

Structure of the manuals

<table>
<thead>
<tr>
<th>Manual</th>
<th>Purpose and usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>FUJITSU Cloud Service K5 IaaS API Reference (this document)</td>
<td>Detailed reference for using the REST API.</td>
</tr>
<tr>
<td>FUJITSU Cloud Service K5 IaaS Features Handbook</td>
<td>Explains the features provided by this service.</td>
</tr>
<tr>
<td>FUJITSU Cloud Service K5 IaaS API User Guide</td>
<td>Explains how to use the REST API, how to build the API runtime environment, and sample scripts according to usage sequences, etc.</td>
</tr>
</tbody>
</table>

Trademarks

- Adobe, the Adobe logo, Acrobat, and Adobe Reader are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.
- Apache and Tomcat are either registered trademarks or trademarks of The Apache Software Foundation in the United States and/or other countries.
- Microsoft, Windows, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
- VMware, the VMware logo and VMotion are registered trademarks or trademarks of VMware, Inc. in the United States and other countries.
- The OpenStack Word Mark is either a registered trademark/service mark or trademark/service mark of the OpenStack Foundation, in the United States and other countries and is used with the OpenStack Foundation’s permission.
- Oracle and Java are registered trademarks of Oracle and/or its affiliates.
- Red Hat is a registered trademark of Red Hat, Inc. in the U.S. and other countries.
- UNIX is a registered trademark of The Open Group.

Note that trademark symbols ("™" or "®") are not appended to system names or product names used in this document.

Export restrictions

Exportation/release of this document may require necessary procedures in accordance with the regulations of your resident country and/or US export control laws.

Notice

- Information in this document may be subject to change without prior notice.
- No part of the content of this document may be reproduced without the written permission of Fujitsu Limited.
- Information in this document is the latest information as of the date of revision.
- Fujitsu assumes no responsibility for infringement of any patent rights or other rights of third parties arising from the use of information in this document.
## Contents

### Part 1: Compute

1. **Location service** ................................................................. 2
   1.1 Generate URLs when using APIs ........................................... 2
   1.2 API list .................................................................................. 2
   1.3 Request header ....................................................................... 2
   1.4 API error codes ..................................................................... 3
   1.5 Notes ...................................................................................... 3
   1.6 API details ............................................................................. 3

   1.1.1 Generate URLs when using APIs ........................................... 2
   1.1.2 API list .................................................................................. 2
   1.1.3 Request header ....................................................................... 2
   1.1.4 API error codes ..................................................................... 3
   1.1.5 Notes ...................................................................................... 3
   1.1.6 API details ............................................................................. 3

2. **Virtual servers** ................................................................. 7
   2.1 Generate URLs when using APIs ........................................... 7
   2.2 API list .................................................................................. 7
   2.3 Request header ....................................................................... 13
   2.4 API error codes ..................................................................... 14
   2.5 Notes ...................................................................................... 14
   2.6 API details ............................................................................. 15

   2.6.1 List servers ........................................................................ 15
   2.6.2 Create server ....................................................................... 16
   2.6.3 Get server details ................................................................ 21
   2.6.4 Update server ..................................................................... 22
   2.6.5 Delete server ....................................................................... 26
   2.6.6 Reboot server ..................................................................... 26
   2.6.7 Rebuild server ................................................................... 27
   2.6.8 Resize server ....................................................................... 29
   2.6.9 Confirm resized server ....................................................... 30
   2.6.10 Revert resized server ......................................................... 31
   2.6.11 List images details ............................................................ 31
   2.6.12 Get image details .............................................................. 36
   2.6.13 Delete image ..................................................................... 37
   2.6.14 Show image metadata ....................................................... 37
   2.6.15 Create or replace image metadata ..................................... 38
   2.6.16 Update image metadata items .......................................... 38
   2.6.17 Show image metadata item details ..................................... 40
   2.6.18 Create or update image metadata item ............................ 40
   2.6.19 Delete image metadata item .............................................. 41
   2.6.20 Create server with scheduler hints .................................... 42
   2.6.21 Create multiple servers .................................................... 47
   2.6.22 Show server extended status ............................................. 52
   2.6.23 List extended status for servers ........................................ 54
   2.6.24 List servers with extended server attributes .................... 55
   2.6.25 Show extended server attributes ...................................... 56
   2.6.26 List servers with IP type ................................................... 57
   2.6.27 Create server ..................................................................... 59
   2.6.28 Show server information .................................................. 63
   2.6.29 List servers ........................................................................ 65
   2.6.30 Get image information ...................................................... 66
   2.6.31 List images ......................................................................... 67
   2.6.32 Create server with configuration drive ............................ 71
   2.6.33 Get server information with configuration drive ............... 76
   2.6.34 Get server details with configuration drive ....................... 77
1.8 Other (common) ................................................................. 181
  1.8.1 Generate URLs when using APIs........................................ 181
  1.8.2 API list............................................................................... 181
  1.8.3 Request header................................................................. 183
  1.8.4 API error codes............................................................... 183
  1.8.5 Notes.................................................................................. 184
  1.8.6 API details......................................................................... 184
      1.8.6.1 List key pairs............................................................ 184
      1.8.6.2 Create or import key pair......................................... 185
      1.8.6.3 Delete key pair......................................................... 186
      1.8.6.4 Show key pair information...................................... 187
      1.8.6.5 Get console output for an instance........................ 187
      1.8.6.6 List details for images............................................. 188
      1.8.6.7 Get image details..................................................... 192
      1.8.6.8 Get server password................................................. 193
      1.8.6.9 Update image.......................................................... 194
      1.8.6.10 List images.............................................................. 195
      1.8.6.11 Get an Image........................................................... 196
      1.8.6.12 Delete image............................................................ 198
      1.8.6.13 Create image member............................................ 198
      1.8.6.14 List image member.................................................. 199
      1.8.6.15 Show image member details................................. 201
      1.8.6.16 Delete image member............................................. 201
      1.8.6.17 Update image member.......................................... 201
      1.8.6.18 Update image (v1)................................................ 202
2.7 Object storage

2.7.1 Generate URLs when using APIs

2.7.2 API list

2.7.3 General requirements

2.7.4 API details

2.7.4.1 List containers

- GET /v1/{account}?limit,marker,end_marker,format,prefix,delimiter

2.7.4.2 Update account metadata

- POST /v1/{account}

2.7.4.3 Retrieve account metadata

- HEAD /v1/{account}

2.7.4.4 List objects

- GET /v1/{account}/{container}?limit,marker,end_marker,prefix,format,delimiter,path

2.7.4.5 Create container

- PUT /v1/{account}/{container}

2.7.4.6 Delete container

- DELETE /v1/{account}/{container}

2.7.4.7 Update container metadata

- POST /v1/{account}/{container}

2.7.4.8 Retrieve container metadata

- HEAD /v1/{account}/{container}

2.7.4.9 Download object

- GET /v1/{account}/{container}/{object}

2.7.4.10 Upload object

- PUT /v1/{account}/{container}/{object}?multipart-manifest,signature,expires

2.7.4.11 Copy object

- COPY /v1/{account}/{container}/{object}

2.7.4.12 Delete object

- DELETE /v1/{account}/{container}/{object}

2.7.4.13 Retrieve object metadata

- HEAD /v1/{account}/{container}/{object}

2.7.4.14 Update object metadata

- POST /v1/{account}/{container}/{object}
Part 1: Compute

Topics:

- Location service
- Virtual servers
- Dedicated instances
- Provisioning script
- Auto scale
- Virtual server imports
- Virtual servers for SAP
- Other (common)
1.1 Location service

1.1.1 Generate URLs when using APIs

The APIs require URLs of the compute type, which can be generated by the identity service on the Service catalog.

The endpoint URL is returned in the following format by the identity service.

https://hostName/v2/{tenant_id}

Create the URL in one of the following formats:
- If you remember the tenant_id:
  Join the path name of each API in the host section of the end point URL
- If you do not remember the tenant_id:
  The endpoint URL and the path name of each API with "v2/{tenant_id}" removed from the beginning

1.1.2 API list

Servers with extended availability zones

<table>
<thead>
<tr>
<th>Item</th>
<th>API name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /v2/{tenant_id}/servers/{server_id} Show server</td>
<td>Displays information of the specified server with availability zones</td>
</tr>
<tr>
<td>2</td>
<td>GET /v2/{tenant_id}/servers/detail List details for servers</td>
<td>Lists details for servers with current availability zones</td>
</tr>
</tbody>
</table>

1.1.3 Request header

<table>
<thead>
<tr>
<th>Header</th>
<th>Specified value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accept</td>
<td>application/json</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>X-Auth-Token</td>
<td>Authentication token</td>
</tr>
</tbody>
</table>
1.1.4 API error codes

<table>
<thead>
<tr>
<th>Error code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>500, other codes possible</td>
<td>Server Error, cloudServersFault</td>
</tr>
<tr>
<td>400</td>
<td>badRequest</td>
</tr>
<tr>
<td>401</td>
<td>unauthorized</td>
</tr>
<tr>
<td>403</td>
<td>Forbidden, resizeNotAllowed</td>
</tr>
<tr>
<td>404</td>
<td>itemNotFound</td>
</tr>
<tr>
<td>405</td>
<td>badMethod</td>
</tr>
<tr>
<td>409</td>
<td>conflictingRequest</td>
</tr>
<tr>
<td>413</td>
<td>overLimit</td>
</tr>
<tr>
<td>415</td>
<td>badMediaType</td>
</tr>
<tr>
<td>501</td>
<td>notImplemented</td>
</tr>
<tr>
<td>503</td>
<td>serviceUnavailable</td>
</tr>
</tbody>
</table>

1.1.5 Notes

When an API (show instance list, etc.) used to display a list of resources is executed, only some of the availability zone information may be returned. If this happens, it is assumed that infrastructure maintenance is in progress, so wait for a few moments (at least one minute) and then execute the API again.

1.1.6 API details

1.1.6.1 Show server

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/{server_id}</td>
<td>Shows information for a specified server, including its availability zone.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the show server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.
Response

Example. Show server: JSON response

```json
{
  "server": {
    "OS-EXT-AZ:availability_zone": "nova",
    "accessIPv4": "",
    "accessIPv6": "",
    "addresses": {
      "private": [
        {
          "addr": "192.168.0.3",
          "version": 4
        }
      ]
    },
    "created": "2013-01-30T13:38:47Z",
    "flavor": {
      "id": "1",
      "links": [
        {
          "href": "http://openstack.example.com/openstack/flavors/1",
          "rel": "bookmark"
        }
      ],
      "hostId": "d38ea49a033b00efaf80c165de63f4805c886dfb94dc0fe731227eccb",
      "id": "fb7babfd-ela1-4add-90e6-3558180983c7",
      "image": {
        "id": "70a599e0-31e7-49b7-b260-868f441e862b",
        "links": [
          {
            "href": "http://openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
            "rel": "bookmark"
          }
        ]
      }
    },
    "hostId": "d38ea49a033b00efaf80c165de63f4805c886dfb94dc0fe731227eccb",
    "id": "fb7babfd-ela1-4add-90e6-3558180983c7",
    "image": {
      "id": "70a599e0-31e7-49b7-b260-868f441e862b",
      "links": [
        {
          "href": "http://openstack.example.com/openstack/v2/openstack/servers/fb7babfd-ela1-4add-90e6-3558180983c7",
          "rel": "self"
        },
        {
          "href": "http://openstack.example.com/openstack/servers/fb7babfd-ela1-4add-90e6-3558180983c7",
          "rel": "bookmark"
        }
      ],
    "metadata": {
      "My Server Name": "Apache1"
    },
    "name": "new-server-test",
    "progress": 0,
    "status": "ACTIVE",
    "tenant_id": "openstack",
    "updated": "2013-01-30T13:38:49Z",
    "user_id": "fake"
  }
}
```
1.1.6.2 List details for servers

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/detail</td>
<td>Lists details for servers, including their current availability zone.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the list details for servers request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. List details for servers: JSON response**

```json
{
    "servers": [
        {
            "OS-EXT-AZ:availability_zone": "nova",
            "accessIPv4": "",
            "accessIPv6": "",
            "addresses": {
                "private": [
                    {
                        "addr": "192.168.0.3",
                        "version": 4
                    }
                ]
            },
            "created": "2013-01-30T13:26:51Z",
            "flavor": {
                "id": "1",
                "links": [ {
                    "href": "http://openstack.example.com/openstack/flavors/1",
                    "rel": "bookmark"
                } ],
            },
            "hostId": "60c988a84401fa15808a32833e5848e9caa99a45778310ba7b363165",
            "id": "3dbf5b00-dabc-41ff-b6ab-4409568fae9d",
            "image": { "id": "70a599e0-31e7-49b7-b260-868f441e862b", "links": [ { "href": "http://openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
                        "rel": "bookmark"
                    } ] },
            "links": [ { } ]
        }
    ]
```
"href": "http://openstack.example.com/v2/openstack/servers/3dbf5b00-dabc-41ff-b6ab-4409568fae9d",
"rel": "self"
},
{
"href": "http://openstack.example.com/openstack/servers/3dbf5b00-dabc-41ff-b6ab-4409568fae9d",
"rel": "bookmark"
},

"metadata": {
  "My Server Name": "Apache1"
},

"name": "new-server-test",
"progress": 0,
"status": "ACTIVE",
"tenant_id": "openstack",
"updated": "2013-01-30T13:26:52Z",
"user_id": "fake"
}
1.2 Virtual servers

1.2.1 Generate URLs when using APIs

The APIs require URLs of the compute type, which can be generated by the identity service on the Service catalog. The endpoint URL is returned in the following format by the identity service.

https://hostName/v2/{tenant_id}

Host portion Path portion

Create the URL in one of the following formats:

- If you remember the tenant_id:
  Join the path name of each API in the host section of the end point URL
- If you do not remember the tenant_id:
  The endpoint URL and the path name of each API with "/v2/{tenant_id}" removed from the beginning

1.2.2 API list

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /v2/{tenant_id}/servers</td>
<td>List servers</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lists the IDs and names of all servers</td>
</tr>
<tr>
<td>2</td>
<td>GET /v2/{tenant_id}/servers{?changes-since,image,flavor,name,status,host}</td>
<td>List servers</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lists the specified server name information</td>
</tr>
<tr>
<td>3</td>
<td>GET /v2/{tenant_id}/servers{?changes-since,image,flavor,name,status,host}</td>
<td>List servers</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lists the specified server ID information</td>
</tr>
<tr>
<td>4</td>
<td>POST /v2/{tenant_id}/servers</td>
<td>Create server</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Creates a server</td>
</tr>
<tr>
<td>5</td>
<td>POST /v2/{tenant_id}/servers</td>
<td>Create server</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Creates a specific server</td>
</tr>
<tr>
<td>6</td>
<td>GET /v2/{tenant_id}/servers/detail{?changes-since,image,flavor,name,status,host}</td>
<td>Get server details</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lists all server details</td>
</tr>
<tr>
<td>Item</td>
<td>API</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>7</td>
<td>GET /v2/{tenant_id}/servers/detail?changes-since,image,flavor,name,status,host</td>
<td>Lists details of servers with specified status</td>
</tr>
<tr>
<td></td>
<td>GET /v2/{tenant_id}/servers/{server_id}</td>
<td>Get server details</td>
</tr>
<tr>
<td>8</td>
<td>GET /v2/{tenant_id}/servers/{server_id}</td>
<td>Retrieves detailed information of the specified server</td>
</tr>
<tr>
<td>9</td>
<td>PUT /v2/{tenant_id}/servers/{server_id}</td>
<td>Update server</td>
</tr>
<tr>
<td>10</td>
<td>PUT /v2/{tenant_id}/servers/{server_id}</td>
<td>Update server</td>
</tr>
<tr>
<td>11</td>
<td>DELETE /v2/{tenant_id}/servers/{server_id}</td>
<td>Delete server</td>
</tr>
</tbody>
</table>

**Server actions**

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>POST /v2/{tenant_id}/servers/{server_id}/action</td>
<td>Reboot server</td>
</tr>
<tr>
<td></td>
<td>POST /v2/{tenant_id}/servers/{server_id}/action</td>
<td>Rebuilds the specified server</td>
</tr>
<tr>
<td>3</td>
<td>POST /v2/{tenant_id}/servers/{server_id}/action</td>
<td>Resize server</td>
</tr>
<tr>
<td>4</td>
<td>POST /v2/{tenant_id}/servers/{server_id}/action</td>
<td>Confirms resized server</td>
</tr>
<tr>
<td>5</td>
<td>POST /v2/{tenant_id}/servers/{server_id}/action</td>
<td>Reverts the resized server to its original size</td>
</tr>
<tr>
<td>6</td>
<td>POST /v2/{tenant_id}/servers/{server_id}/action</td>
<td>Shelves the specified server</td>
</tr>
<tr>
<td>7</td>
<td>POST /v2/{tenant_id}/servers/{server_id}/action</td>
<td>Restores a server specified for shelving</td>
</tr>
</tbody>
</table>
## Images

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /v2/{tenant_id}/images/detail{?changes-since,server,name,status,type}</td>
<td>Lists the details of available images</td>
</tr>
<tr>
<td>2</td>
<td>GET /v2/{tenant_id}/images/detail{?changes-since,server,name,status,type}</td>
<td>Lists the details of available specified images</td>
</tr>
<tr>
<td>3</td>
<td>GET /v2/{tenant_id}/images/{image_id}</td>
<td>Retrieves the details of the specified image</td>
</tr>
<tr>
<td>4</td>
<td>DELETE /v2/{tenant_id}/images/{image_id}</td>
<td>Deletes the specified image</td>
</tr>
</tbody>
</table>

## Metadata

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /v2/{tenant_id}/images/{image_id}/metadata</td>
<td>Displays the metadata of the specified images</td>
</tr>
<tr>
<td>2</td>
<td>POST /v2/{tenant_id}/images/{image_id}/metadata</td>
<td>Creates or replaces metadata for the specified image</td>
</tr>
<tr>
<td>3</td>
<td>PUT /v2/{tenant_id}/images/{image_id}/metadata</td>
<td>Updates the metadata of images identified by key</td>
</tr>
<tr>
<td>4</td>
<td>GET /v2/{tenant_id}/images/{image_id}/metadata/{key}</td>
<td>Displays the metadata item (identified by key) of the specified image</td>
</tr>
<tr>
<td>5</td>
<td>PUT /v2/{tenant_id}/images/{image_id}/metadata/{key}</td>
<td>Creates or updates the metadata item identified by key</td>
</tr>
<tr>
<td>6</td>
<td>DELETE /v2/{tenant_id}/images/{image_id}/metadata/{key}</td>
<td>Deletes the metadata item identified by key</td>
</tr>
</tbody>
</table>

## Servers with scheduler hints

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>POST /v2/{tenant_id}/servers</td>
<td>Creates a server using scheduler instructions that were passed directly to the scheduler.</td>
</tr>
</tbody>
</table>
### Server multiple creation

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>POST /v2/{tenant_id}/servers</td>
<td>Creates one or more servers that have the optional booking ID</td>
</tr>
<tr>
<td>2</td>
<td>POST /v2/{tenant_id}/servers</td>
<td>Creates one or more servers with the optional minimum count</td>
</tr>
<tr>
<td>3</td>
<td>POST /v2/{tenant_id}/servers</td>
<td>Creates one or more servers with the optional maximum count</td>
</tr>
</tbody>
</table>

### Server extended status

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /v2/{tenant_id}/servers/{server_id}</td>
<td>Displays the extended status attributes of specified servers</td>
</tr>
<tr>
<td>2</td>
<td>GET /v2/{tenant_id}/servers/detail</td>
<td>Lists the details of the extended status attributes of all servers</td>
</tr>
</tbody>
</table>

### Server extended attributes

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /v2/{tenant_id}/servers</td>
<td>Lists the details of the extended server attributes of all servers</td>
</tr>
<tr>
<td>2</td>
<td>GET /v2/{tenant_id}/servers/{server_id}</td>
<td>Displays the extended server attributes of specified servers</td>
</tr>
</tbody>
</table>

### Server IP type

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /v2/{tenant_id}/servers/detail</td>
<td>Lists the fixed and floating IP addresses by type for all servers</td>
</tr>
</tbody>
</table>

### Servers and images with disk config

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>POST /v2/{tenant_id}/servers</td>
<td>Creates a server</td>
</tr>
<tr>
<td>2</td>
<td>GET /v2/{tenant_id}/servers/{server_id}</td>
<td>Displays information about the specified server</td>
</tr>
<tr>
<td>3</td>
<td>GET /v2/{tenant_id}/servers/detail</td>
<td>Lists the servers</td>
</tr>
<tr>
<td>Item</td>
<td>API</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>-------------</td>
</tr>
<tr>
<td>4</td>
<td>GET /v2/{tenant_id}/images/{image_id}</td>
<td>Retrieves information about the specified image</td>
</tr>
<tr>
<td>5</td>
<td>GET /v2/{tenant_id}/images/detail</td>
<td>Lists the images</td>
</tr>
</tbody>
</table>

### Configuration drive

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>POST /v2/{tenant_id}/servers</td>
<td>Creates a server using extended attributes of the driver settings</td>
</tr>
<tr>
<td>2</td>
<td>GET /v2/{tenant_id}/servers/{server_id}</td>
<td>Displays information about the specified server, including extended attributes of the driver settings</td>
</tr>
<tr>
<td>3</td>
<td>GET /v2/{tenant_id}/servers/detail</td>
<td>Displays detailed information about all servers, including extended attributes of the driver settings</td>
</tr>
</tbody>
</table>

### Server OS-EXT-IPS-MAC:mac_addr extended attribute

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>POST /v2/{tenant_id}/servers</td>
<td>Creates a server using OS-EXT-IPS-MAC:mac_addr</td>
</tr>
<tr>
<td>2</td>
<td>GET /v2/{tenant_id}/servers/{server_id}</td>
<td>Displays information about the specified server (including OS-EXT-IPS-MAC:mac_addr)</td>
</tr>
<tr>
<td>3</td>
<td>GET /v2/{tenant_id}/servers/detail</td>
<td>Lists details of all servers (including OS-EXT-IPS-MAC:mac_addr)</td>
</tr>
</tbody>
</table>

### Servers with block device mapping format

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>POST /v2/{tenant_id}/servers</td>
<td>Creates a server using block device mapping</td>
</tr>
<tr>
<td>2</td>
<td>GET /v2/{tenant_id}/servers?changes-since,image,flavor,name,status,host</td>
<td>Lists the ID, name and link of all servers</td>
</tr>
</tbody>
</table>
### Volume attachment

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | POST /v2/{tenant_id}/servers/{server_id}/os-volume_attachments | Attaches a volume to the specified server
|      | Attach volume | |
| 2    | GET /v2/{tenant_id}/servers/{server_id}/os-volume_attachments | Lists the volumes of the specified server
|      | List volume attachments | |
| 3    | GET /v2/{tenant_id}/servers/{server_id}/os-volume_attachments/{attachment_id} | Displays details of the specified volume
|      | Show volume attachment details | |
| 4    | DELETE /v2/{tenant_id}/servers/{server_id}/os-volume_attachments/{attachment_id} | Detaches the specified volume from the specified server
|      | Delete volume attachment | |

### Server start and stop

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | POST /v2/{tenant_id}/servers/{server_id}/action | Sets the status to ACTIVE, and starts the server
|      | Start server | |
| 2    | POST /v2/{tenant_id}/servers/{server_id}/action | Sets the status to STOPPED, and stops the server
|      | Stop server | |

### Flavor access

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | GET /v2/{tenant_id}/flavors | Lists the flavors and access types
|      | List flavors with access type | |
| 2    | GET /v2/{tenant_id}/flavors/{flavor_id} | Retrieves the flavor access type
|      | Show flavor access type | |

### Attach interfaces

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | POST /v2/{tenant_id}/servers/{server_id}/os-interface | Creates and uses a port for connecting to a server instance
<p>|      | Create interface | |
| 2    | GET /v2/{tenant_id}/servers/{server_id}/os-interface | Lists the ports |</p>
<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>GET /v2/{tenant_id}/servers/{server_id}/os-interface/{attachment_id}</td>
<td>Displays information related to the specified port</td>
</tr>
<tr>
<td></td>
<td>Show attached interface information</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>DELETE /v2/{tenant_id}/servers/{server_id}/os-interface/{attachment_id}</td>
<td>Detaches the specified port</td>
</tr>
<tr>
<td></td>
<td>Detach interface</td>
<td></td>
</tr>
</tbody>
</table>

### Server groups (os-server-groups)

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /v2/{tenant_id}/os-server-groups</td>
<td>Lists the server groups</td>
</tr>
<tr>
<td></td>
<td>Lists server groups</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>POST /v2/{tenant_id}/os-server-groups</td>
<td>Creates a server group</td>
</tr>
<tr>
<td></td>
<td>Creates a server group</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>GET /v2/{tenant_id}/os-server-groups/{ServerGroup_id}</td>
<td>Retrieves detailed information about the specified server group</td>
</tr>
<tr>
<td></td>
<td>Shows server group details</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>DELETE /v2/{tenant_id}/os-server-groups/{ServerGroup_id}</td>
<td>Deletes the server group</td>
</tr>
<tr>
<td></td>
<td>Deletes server group</td>
<td></td>
</tr>
</tbody>
</table>

### Server metadata

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>POST /v2/{tenant_id}/servers/{server_id}/metadata</td>
<td>Updates the metadata item (identified by key) of the specified server</td>
</tr>
<tr>
<td></td>
<td>Update server metadata items</td>
<td></td>
</tr>
</tbody>
</table>

### 1.2.3 Request header

<table>
<thead>
<tr>
<th>Header</th>
<th>Specified value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accept</td>
<td>application/json</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>X-Auth-Token</td>
<td>Valid authentication token</td>
</tr>
</tbody>
</table>
1.2.4 API error codes

<table>
<thead>
<tr>
<th>Error code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>500, other codes possible</td>
<td>Server Error, cloudServersFault</td>
</tr>
<tr>
<td>400</td>
<td>badRequest</td>
</tr>
<tr>
<td>401</td>
<td>unauthorized</td>
</tr>
<tr>
<td>403</td>
<td>Forbidden, resizeNotAllowed</td>
</tr>
<tr>
<td>404</td>
<td>itemNotFound</td>
</tr>
<tr>
<td>405</td>
<td>badMethod</td>
</tr>
<tr>
<td>409</td>
<td>conflictingRequest, buildInProgress</td>
</tr>
<tr>
<td>413</td>
<td>overLimit</td>
</tr>
<tr>
<td>415</td>
<td>badMediaType</td>
</tr>
<tr>
<td>422</td>
<td>HTTPUnprocessableEntity</td>
</tr>
<tr>
<td>501</td>
<td>notImplemented</td>
</tr>
<tr>
<td>503</td>
<td>serviceUnavailable</td>
</tr>
</tbody>
</table>

1.2.5 Notes

- When an API (show instance list, etc.) used to display a list of resources is executed, only some of the availability zone information may be returned. If this happens, it is assumed that infrastructure maintenance is in progress, so wait for a few moments (at least one minute) and then execute the API again.

- Description of API error code

  The response code 500 is returned when execution of the APIs below is not approved under the operator privileges (role).
  - GET /v2/{tenant_id}/images/{image_id}
  - GET /v2/{tenant_id}/images/{image_id}/metadata
  - GET /v2/{tenant_id}/images/{image_id}/metadata/{key}
  - POST /v2/{tenant_id}/images/{image_id}/metadata
  - PUT /v2/{tenant_id}/images/{image_id}/metadata
  - PUT /v2/{tenant_id}/images/{image_id}/metadata/{key}
  - DELETE /v2/{tenant_id}/images/{image_id}/metadata/{key}
1.2.6 API details

1.2.6.1 List servers

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers{?changes-since,image,flavor,name,status,host}</td>
<td>Lists IDs, names, and links for all servers.</td>
</tr>
</tbody>
</table>

Normal response codes: 200, 203

**Request**

This table shows the URI parameters for the list servers request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This table shows the query parameters for the list servers request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>changes-since</td>
<td>DateTime</td>
<td>A time/date stamp for when the server last changed status.</td>
</tr>
<tr>
<td>image</td>
<td>AnyURI</td>
<td>Name of the image in URL format.</td>
</tr>
<tr>
<td>flavor</td>
<td>AnyURI</td>
<td>Name of the flavor in URL format.</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the server as a string.</td>
</tr>
<tr>
<td>status</td>
<td>Server Status</td>
<td>Value of the status of the server so that you can filter on &quot;ACTIVE&quot; for example.</td>
</tr>
<tr>
<td>host</td>
<td>String</td>
<td>Name of the host as a string.</td>
</tr>
</tbody>
</table>

**Response**

**Example. List servers: JSON response**

```json
{
  "servers": [
    {
      "id": "616fb98f-46ca-475e-917e-2563e5a8cd19",
      "links": [
        {
          "href": "http://openstack.example.com/v2/openstack/servers/616fb98f-46ca-475e-917e-2563e5a8cd19",
          "rel": "bookmark"
        }
      ]
    }
  ]
}
```
1.2.6.2 Create server

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers</td>
<td>Creates a server.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

**Caution**
- When an instance created using a Linux image is started or restarted, the sshd settings are initialized, and it may no longer be possible to log in to the instance. Refer to "Reboot server" on page 26 for details on the required tasks.
- When the password of the user who registered (imported) the image is changed, the image can no longer be used. In such a case, convey either of the following to the operator and request a change of image:
  - The location of the image following the change of image UUID and user authentication information
  - The image UUID that was registered (imported) following the change of image UUID and user authentication information

**Request**

This table shows the URI parameters for the create server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the create server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>security_group</td>
<td>string</td>
<td>One or more security_group objects. Specify the name of the security group in the name attribute. If you omit this attribute, the server is created in the default security group. Specify a security group for which TCP communication is permitted for the following IP address and port number. IP address: 169.254.169.254 Port number: 80 If TCP communication is not permitted, the host name (computer name) and administrator</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>password may not be set when creating an instance.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| user_data     | string (Optional) | Configuration information or scripts to use upon launch. Must be Base64 encoded. The main formats that are supported are as follows:  
  • Linux:  
    • Shell script (begins with #!)  
  • Windows:  
    • PowerShell (begins with #ps1_sysnative or #ps1_x86)  
    • Windows batch (begins with rem cmd)  
  If Linux, cloud-config files can also be specified, but since verification has not been completed, we recommend using Shell scripts. |
| availability_zone | string (Optional) | The availability zone in which to launch the server. |
| server        | ServerForCreate | server. |
| imageRef      | String       | The image reference for the desired image for your server instance. Specify as an ID or full URL. |
| flavorRef     | String       | The flavor reference for the desired flavor for your server instance. Specify as an ID or full URL. |
| key_name      | string (Optional) | Assigns the public key of the named keypair to the server. |
| networks      | string (Optional) | A networks object. By default, the server instance is provisioned with all isolated networks for the tenant.  
  Optionally, you can create one or more NICs on the server.  
  To provision the server instance with a NIC for a network, specify the UUID of the network in the uuid attribute in a networks object.  
  To provision the server instance with a NIC for an already existing port, specify the port-id in the port attribute in a networks object.  
  You can specify multiple NICs on the server.  
  Ensure that the network is specified.  
  To set the following information for a virtual server, a virtual router must be connected to the network that the virtual server will be connected to.  
  • Host name (computer name) |
<p>| server        | ServerForCreate | server. |</p>
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
</table>
|            |               | • Administrator password  
|            |               | • Authentication key (key pair)                                                                                                           |
| uuid       | string (Optional) | To provision the server instance with a NIC for a network, specify the UUID of the network in the uuid attribute in a networks object. Required if you omit the port attribute.  
|            |               | If the network includes multiple subnets, IP addresses will be allocated from any subnet. To allocate an IP address of a specific subnet, create a port in advance, and specify the uuid of that port in port. |
| port       | string (Optional) | To provision the server instance with a NIC for an already existing port, specify the port-id in the port attribute in a networks object. Required if you omit the uuid attribute. |
| fixed_ip   | string (Optional) | A fixed IPv4 address for the NIC. Valid with a neutron or nova-networks network.                                                             |
| name       | string        | The server name.  
|            |               | This information is also used as the computer name/host name.  
|            |               | If 64 characters or more are specified:  
|            |               | • Windows:  
|            |               | The computer name is the default name set by Windows.  
|            |               | • Linux:  
|            |               | The host name will be "host-fixedIpAddressOfEth0".                                                                                  |
|            |               | The string set for the computer name/host name is changed as follows:  
|            |               | • Spaces ( ) and underscores (_) are replaced with hyphens (-)  
|            |               | • Uppercase letters are replaced with lowercase letters  
|            |               | • Symbols other than periods (.) and hyphens (-) are removed  
|            |               | • Periods (.) are removed from the beginning and end of the string, and strings consisting of hyphens (-) are removed  
|            |               | • In Windows, if the string contains a period (.) anywhere other than at the beginning or end, then the characters preceding the period (.) will be used for the name. |
| metadata   | string (Optional) | Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each.  
|            |               | • Windows:  
<p>|            |               | To specify a password of an instance, specify the &quot;admin_pass&quot; key. |</p>
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>block_device_mapping_v2</td>
<td>string</td>
<td>Enables booting the server from a volume when additional parameters are given.</td>
</tr>
<tr>
<td>device_name</td>
<td>string</td>
<td>Describes a path to the device for the volume you want to use to boot the server. Specify this item in /dev/vd&lt;deviceName&gt; format. /dev/vd is fixed, and for &lt;deviceName&gt;, specify characters that are valid as a device name. When creating an instance that is allocated multiple volumes, for the boot volume, specify the character with highest priority among the device names of all volumes. The order of priority is a &gt; b &gt; c &gt; ...</td>
</tr>
<tr>
<td>source_type</td>
<td>string</td>
<td>Describes the volume source type for the volume. Choices are &quot;snapshot&quot;, &quot;volume&quot;, or &quot;image&quot;.</td>
</tr>
<tr>
<td>destination_type</td>
<td>string</td>
<td>Specifies the connection destination (&quot;volume&quot;).</td>
</tr>
<tr>
<td>delete_on_termination</td>
<td>bool</td>
<td>Specifies whether volumes created during the instance creation will be deleted when the instance is deleted. When &quot;True&quot; is specified, volumes during instance creation will also be deleted when the instance is deleted. When &quot;False&quot; is specified, the volumes created during instance creation will not be deleted when the instance is deleted. If not specified, &quot;False&quot; (do not delete) will be used. The volume where snapshots are collected will not be deleted even if &quot;True&quot; is specified.</td>
</tr>
<tr>
<td>boot_index</td>
<td>String</td>
<td>Specifies the device start order. Specify sequential values, starting from 0. For the boot disk, specify &quot;0&quot;.</td>
</tr>
<tr>
<td>config_drive</td>
<td>String</td>
<td>Only &quot;false&quot; can be specified.</td>
</tr>
<tr>
<td>uuid</td>
<td>uuid</td>
<td>Specifies the UUID of the resource specified for source_type.</td>
</tr>
<tr>
<td>volume_size</td>
<td>String</td>
<td>Specify the volume size in GB. This item must be specified when &quot;image&quot; is specified for source_type. Specify a value equal to the volume size in GB.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>to or higher than the min_disk parameter of the image to be used. If the min_disk parameter of the image to be used has not been specified or is &quot;0&quot;, check the minimum size with the image provider and specify the value accordingly.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If &quot;volume&quot; was specified for source_type, this item will be ignored even if a value is specified.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If &quot;snapshot&quot; was specified for source_type, and this item is omitted, the volume size of the snapshot collection source will be used.</td>
</tr>
</tbody>
</table>

**Example. Create server: JSON request**

```json
{
  "server": {
    "name": "server-test-1",
    "imageRef": "b5660a6e-4b46-4be3-9707-6b47221b454f",
    "flavorRef": "2",
    "key_name": "keypair1",
    "networks": [
      {
        "uuid": "d32019d3-bc6e-4319-9c1d-6722fc136a22"
      },
      {
        "port": "2f2eab14-5c2f-4111-871f-f752c73ca3bf"
      }
    ],
    "security_groups": [
      {
        "name": "default"
      },
      {
        "name": "another-secgroup-name"
      }
    ],
    "block_device_mapping_v2": [
      {
        "device_name": "/dev/vda",
        "source_type": "image",
        "destination_type": "volume",
        "volume_size": "20",
        "boot_index": "0",
        "uuid": "6cbf9710-87e3-4a36-8116-9b3396882621",
        "delete_on_termination": "True"
      },
      {
        "device_name": "/dev/vdb",
        "source_type": "volume",
        "destination_type": "volume",
        "boot_index": "1",
        "uuid": "0a273d8d-c5e1-4886-bd93-1d1779283fa3",
        "delete_on_termination": "True"
      },
      {
        "device_name": "/dev/vdc",
        "source_type": "snapshot",
        "destination_type": "volume",
        "volume_size": "30",
        "boot_index": "2",
        "uuid": "492eac4d-6c12-4828-b0ec-75d3bff0bd4b",
        "delete_on_termination": "True"
      }
    ]
  }
}```
Response

Example. Create server: JSON response

```json
{
  "server": {
    "security_groups": [
      {
        "name": "default"
      }
    ],
    "OS-DCF:diskConfig": "MANUAL",
    "id": "c6d04159-9bfc-4ab8-823d-0d5ca2abe152",
    "links": [
      {
        "href": "http://166.78.46.130:8774/v2/4fd44f30292945e481c7b8a0c8908869/servers/c6d04159-9bfc-4ab8-823d-0d5ca2abe152",
        "rel": "self"
      },
      {
        "href": "http://166.78.46.130:8774/4fd44f30292945e481c7b8a0c8908869/servers/c6d04159-9bfc-4ab8-823d-0d5ca2abe152",
        "rel": "bookmark"
      }
    ],
    "adminPass": "aabbccddeeff"
  }
}
```

1.2.6.3 Get server details

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/{server_id}</td>
<td>Gets details for a specified server.</td>
</tr>
</tbody>
</table>

Normal response codes: 200, 203

Request

This table shows the URI parameters for the get server details request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. Get server details: JSON response
1.2.6.4 Update server

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUT</td>
<td>/v2/{tenant_id}/servers/{server_id}</td>
<td>Updates the editable attributes of the specified server.</td>
</tr>
</tbody>
</table>
Normal response codes: 200

### Request

This table shows the URI parameters for the update server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server.</td>
</tr>
</tbody>
</table>

#### Example. Update Server Name Request: JSON

```json
{
  "server": {
    "name": "new-server-test"
  }
}
```

#### Example. Update Server IP Addresses Request: JSON

```json
{
  "server": {
    "accessIPv4": "67.23.10.132",
    "accessIPv6": "::babe:67.23.10.132"
  }
}
```

### Response

#### Example. Update Server Name Response: JSON

```json
{
  "server": {
    "id": "52415800-8b69-11e0-9b19-734f565bc83b",
    "tenant_id": "1234",
    "user_id": "5678",
    "name": "new-server-test",
    "created": "2010-11-11T12:00:00Z",
    "updated": "2010-11-12T12:44:44Z",
    "hostId": "e4d909c290d0fblca068ffaddf22cb0",
    "accessIPv4": "67.23.10.138",
    "accessIPv6": "::babe:67.23.10.138",
    "progress": 0,
    "status": "ACTIVE",
    "image": {
      "id": "52415800-8b69-11e0-9b19-734f6f006e54",
      "name": "CentOS 5.2",
      "links": [
        {
          "rel": "self",
          "href": "http://servers.api.openstack.org/v2/1234/images/52415800-8b69-11e0-9b19-734f6f006e54"
        },
        {
          "rel": "bookmark",
          "href": "http://servers.api.openstack.org/1234/images/52415800-8b69-11e0-9b19-734f6f006e54"
        }
```
Example. Update Server IP Addresses Response: JSON

```json
{
    "server": {
        "id": "52415800-8b69-11e0-9b19-734f565bc83b",
        "tenant_id": "1234",
        "flavor": {
            "id": "52415800-8b69-11e0-9b19-734f1195ff37",
            "name": "256 MB Server",
            "links": [
                {
                    "rel": "self",
                    "href": "http://servers.api.openstack.org/v2/1234/flavors/52415800-8b69-11e0-9b19-734f1195ff37"
                },
                {
                    "rel": "bookmark",
                    "href": "http://servers.api.openstack.org/1234/flavors/52415800-8b69-11e0-9b19-734f1195ff37"
                }
            ]
        },
        "metadata": {
            "My Server Name": "Apache1"
        },
        "addresses": {
            "public": [
                {
                    "version": 4,
                    "addr": "67.23.10.138"
                },
                {
                    "version": 6,
                    "addr": "::babe:67.23.10.138"
                }
            ],
            "private": [
                {
                    "version": 4,
                    "addr": "10.176.42.19"
                },
                {
                    "version": 6,
                    "addr": "::babe:10.176.42.19"
                }
            ],
            "links": [
                {
                    "rel": "self",
                    "href": "http://servers.api.openstack.org/v2/1234/servers/52415800-8b69-11e0-9b19-734f565bc83b"
                },
                {
                    "rel": "bookmark",
                    "href": "http://servers.api.openstack.org/1234/servers/52415800-8b69-11e0-9b19-734f565bc83b"
                }
            ]
        }
    }
}
```
"user_id": "5678",
"name": "new-server-test",
"created": "2010-11-11T12:00:00Z",
"updated": "2010-11-12T12:55:55Z",
"hostId": "e4d909c290d0fb1ca068ffad3f22cb0",
"accessIPv4": "67.23.10.132",
"accessIPv6": "::babe:67.23.10.132",
"progress": 0,
"status": "ACTIVE",
"image": {
    "id": "52415800-8b69-11e0-9b19-734f6f006e54",
    "name": "CentOS 5.2",
    "links": [
      {
        "rel": "self",
        "href": "http://servers.api.openstack.org/v2/1234/images/52415800-8b69-11e0-9b19-734f6f006e54"
      },
      {
        "rel": "bookmark",
        "href": "http://servers.api.openstack.org/1234/images/52415800-8b69-11e0-9b19-734f6f006e54"
      }
    ]
  },
  "flavor": {
    "id": "52415800-8b69-11e0-9b19-734f1195ff37",
    "name": "256 MB Server",
    "links": [
      {
        "rel": "self",
        "href": "http://servers.api.openstack.org/v2/1234/flavors/52415800-8b69-11e0-9b19-734f1195ff37"
      },
      {
        "rel": "bookmark",
        "href": "http://servers.api.openstack.org/1234/flavors/52415800-8b69-11e0-9b19-734f1195ff37"
      }
    ]
  },
  "metadata": {
    "My Server Name": "Apache1"
  },
  "addresses": {
    "public": [
      {
        "version": 4,
        "addr": "67.23.10.138"
      },
      {
        "version": 6,
        "addr": "::babe:67.23.10.138"
      }
    ],
    "private": [
      {
        "version": 4,
        "addr": "10.176.42.19"
      },
      {
        "version": 6,
        "addr": "::babe:10.176.42.19"
      }
    ]
  },
  "links": [  
    {
      "rel": "self",
      "href": "http://servers.api.openstack.org/v2/1234/servers/5678"
    },
    {
      "rel": "bookmark",
      "href": "http://servers.api.openstack.org/1234/servers/5678"
    }  
]
1.2.6.5 Delete server

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DELETE</td>
<td>/v2/{tenant_id}/servers/{server_id}</td>
<td>Deletes a specified server.</td>
</tr>
</tbody>
</table>

Normal response codes: 204

**Request**

This table shows the URI parameters for the delete server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server.</td>
</tr>
</tbody>
</table>

This operation does not accept a request body and does not return a response body.

1.2.6.6 Reboot server

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers/{server_id}/action</td>
<td>Reboots the specified server. Specify the reboot action in the request body.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

**Caution**

- When an instance created using a Linux image is restarted, the sshd settings are initialized, and it may no longer be possible to log in to the instance. If the sshd settings have been changed, execute the following commands with administrator privileges before restarting the instance.

  ```bash
  # chkconfig cloud-init-local off
  # chkconfig cloud-init off
  # chkconfig cloud-config off
  # chkconfig cloud-final off
  ```

- If retrieval of metadata fails, the initial user password may be locked.

  By configuring the setting below, the password will no longer be locked from the next restart.

  - Deploy the cloud-init configuration file.
Request

This table shows the URI parameters for the reboot server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the reboot server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>string</td>
<td>Specifies SOFT or HARD reboot. A SOFT reboot signals the operating system to restart, which allows for graceful shutdown of all processes. A HARD reboot is equivalent to power cycling the server. This parameter is ignored in Compute where a HARD reboot is always performed.</td>
</tr>
</tbody>
</table>

Example. Reboot server: JSON request

```
{
  "reboot": {
    "type": "SOFT"
  }
}
```

Response

This operation does not return a response body.

1.2.6.7 Rebuild server

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers/{server_id}/action</td>
<td>Rebuilds the specified server. Specify the rebuild action in the request body.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

Request

This table shows the URI parameters for the rebuild server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server.</td>
</tr>
</tbody>
</table>
This table shows the body parameters for the rebuild server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>string</td>
<td>The name for the new server.</td>
</tr>
<tr>
<td>metadata</td>
<td>string</td>
<td>Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each.</td>
</tr>
</tbody>
</table>

**Example. Rebuild server: JSON request**

```json
{
  "rebuild": {
    "name": "foobar",
    "adminPass": "seekr3t",
    "accessIPv4": "1.2.3.4",
    "accessIPv6": "fe80::100",
    "metadata": {
      "meta var": "meta val"
    }
  }
}
```

**Response**

This table shows the header parameters for the rebuild server response:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>AnyURI (Required)</td>
<td>Specific URL of the server you want to rebuild.</td>
</tr>
</tbody>
</table>

**Example. Rebuild server: JSON response**

```json
{
  "server": {
    "accessIPv4": "1.2.3.4",
    "accessIPv6": "fe80::100",
    "addresses": {
      "private": [
        {
          "addr": "192.168.0.3",
          "version": 4
        }
      ]
    },
    "adminPass": "seekr3t",
    "created": "2012-09-12T17:20:36Z",
    "flavor": {
      "id": "1",
      "links": [
        {
          "href": "http://openstack.example.com/openstack/flavors/1",
          "rel": "bookmark"
        }
      ]
    },
    "hostId": "1e3da81662354c25560b7e5ea6d8123031f67168b6992f20bb84df69",
    "id": "075e40fe-9f03-4652-ba8e-5f8e2547899a",
    "image": {
      "id": "...
```
1.2.6.8 Resize server

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers/{server_id}/action</td>
<td>Resizes the specified server. Specify the resize action in the request body.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

⚠️ Caution

- To confirm changes, execute "Confirm resized server" after changes are completed.
  1. Check that the server status is VERIFY_RESIZE.
  2. Execute "Confirm resized server".

Refer to "Get server details" on page 21 for details on the API for checking the server status. Refer to "Confirm resized server" on page 30 for details on how to confirm a resize action.

- If retrieval of metadata fails, the initial user password may be locked.

By configuring the setting below, the password will no longer be locked from the next restart.

- Deploy the cloud-init configuration file.

```
# cat << EOF > /etc/cloud/cloud.cfg.d/datasource.cfg
datasource_list: ['OpenStack']
EOF
```
Request
This table shows the URI parameters for the resize server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the resize server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>flavorRef</td>
<td>string</td>
<td>Reference to the flavor.</td>
</tr>
</tbody>
</table>

Example. Resize server: JSON request

```
{
  "resize": {
    "flavorRef": "2"
  }
}
```

Response
This operation does not return a response body.

1.2.6.9 Confirm resized server

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers/{server_id}/action</td>
<td>Confirms a pending resize action. Specify the confirmResize action in the request body.</td>
</tr>
</tbody>
</table>

Normal response codes: 204

Request
This table shows the URI parameters for the confirm resized server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server.</td>
</tr>
</tbody>
</table>

Example. Confirm resized server: JSON request

```
{
  "confirmResize": null
}
```
1.2.6.10 Revert resized server

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers/{server_id}/action</td>
<td>Cancels and reverts a pending resize action. Specify the revertResize action in the request body.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

Request

This table shows the URI parameters for the revert resized server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server.</td>
</tr>
</tbody>
</table>

Example. Revert resized server: JSON request

```json
{
  "revertResize": null
}
```

Response

This operation does not return a response body.

1.2.6.11 List images details

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/images/detail{?changes-since,server,name,status,type}</td>
<td>Lists all details for available images.</td>
</tr>
</tbody>
</table>

Normal response codes: 200, 203

Request

This table shows the URI parameters:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This table shows the query parameters for the list images details request:
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>changes-since</td>
<td>DateTime (Optional)</td>
<td>A time/date stamp for when the image last changed status.</td>
</tr>
<tr>
<td>server</td>
<td>AnyURI (Optional)</td>
<td>Name of the server in URL format.</td>
</tr>
<tr>
<td>name</td>
<td>String (Optional)</td>
<td>Name of the image as a string.</td>
</tr>
<tr>
<td>status</td>
<td>Image Status (Optional)</td>
<td>Value of the status of the image so that you can filter on &quot;ACTIVE&quot; for example.</td>
</tr>
<tr>
<td>type</td>
<td>String (Optional)</td>
<td>Value of the type of image, such as BASE, SERVER, or ALL. Possible values: BASE, SERVER, ALL. Default: ALL.</td>
</tr>
</tbody>
</table>

**Response**

**Example. List images details: JSON response**

```json
{
   "images": [
      {
         "created": "2011-01-01T01:02:03Z",
         "id": "70a599e0-31e7-49b7-b260-868f441e862b",
         "links": [
            {
               "href": "http://openstack.example.com/v2/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
               "rel": "self"
            },
            {
               "href": "http://openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
               "rel": "bookmark"
            },
            {
               "href": "http://glance.openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
               "rel": "alternate",
               "type": "application/vnd.openstack.image"
            }
         ],
         "metadata": {
            "architecture": "x86_64",
            "auto_disk_config": "True",
            "kernel_id": "nokernel",
            "ramdisk_id": "nokernel"
         },
         "minDisk": 0,
         "minRam": 0,
         "name": "fakeimage7",
         "progress": 100,
         "status": "ACTIVE",
         "updated": "2011-01-01T01:02:03Z"
      }
   ]
}```
"created": "2011-01-01T01:02:03Z",
"id": "155d900f-4e14-4e4c-a73d-069cbf4541e6",
"links": [
  {
    "href": "http://openstack.example.com/v2/openstack/images/155d900f-4e14-4e4c-a73d-069cbf4541e6",
    "rel": "self"
  },
  {
    "href": "http://openstack.example.com/openstack/images/155d900f-4e14-4e4c-a73d-069cbf4541e6",
    "rel": "bookmark"
  },
  {
    "href": "http://glance.openstack.example.com/openstack/images/155d900f-4e14-4e4c-a73d-069cbf4541e6",
    "rel": "alternate",
    "type": "application/vnd.openstack.image"
  }
],
"metadata": {
  "architecture": "x86_64",
  "kernel_id": "nokernel",
  "ramdisk_id": "nokernel"
},
"minDisk": 0,
"minRam": 0,
"name": "fakeimage123456",
"progress": 100,
"status": "ACTIVE",
"updated": "2011-01-01T01:02:03Z"
},
{
  "created": "2011-01-01T01:02:03Z",
  "id": "a2459075-d96c-40d5-893e-577ff92e721c",
  "links": [
    {
      "href": "http://openstack.example.com/v2/openstack/images/a2459075-d96c-40d5-893e-577ff92e721c",
      "rel": "self"
    },
    {
      "href": "http://openstack.example.com/openstack/images/a2459075-d96c-40d5-893e-577ff92e721c",
      "rel": "bookmark"
    },
    {
      "href": "http://glance.openstack.example.com/openstack/images/a2459075-d96c-40d5-893e-577ff92e721c",
      "rel": "alternate",
      "type": "application/vnd.openstack.image"
    }
  ],
  "metadata": {
    "kernel_id": "nokernel",
    "ramdisk_id": "nokernel"
  },
  "minDisk": 0,
  "minRam": 0,
  "name": "fakeimage123456",
  "progress": 100,
  "status": "ACTIVE",
  "updated": "2011-01-01T01:02:03Z"
},
{
  "created": "2011-01-01T01:02:03Z",
  "id": "a440c04b-79fa-479c-bed1-0b816eaec379",
  "links": [


```json
{
   "href": "http://openstack.example.com/v2/openstack/images/a440c04b-79fa-479c-bed1-0b816eaec379",
   "rel": "self"
},
{
   "href": "http://openstack.example.com/openstack/images/a440c04b-79fa-479c-bed1-0b816eaec379",
   "rel": "bookmark"
},
{
   "href": "http://glance.openstack.example.com/openstack/images/a440c04b-79fa-479c-bed1-0b816eaec379",
   "rel": "alternate",
   "type": "application/vnd.openstack.image"
}
],
"metadata": {
   "architecture": "x86_64",
   "auto_disk_config": "False",
   "kernel_id": "nokernel",
   "ramdisk_id": "nokernel"
},
"minDisk": 0,
"minRam": 0,
"name": "fakeimage6",
"progress": 100,
"status": "ACTIVE",
"updated": "2011-01-01T01:02:03Z"
},
{
   "created": "2011-01-01T01:02:03Z",
   "id": "c905cedb-7281-47e4-8a62-f26bc5fc4c77",
   "links": [
      {
         "href": "http://openstack.example.com/v2/openstack/images/c905cedb-7281-47e4-8a62-f26bc5fc4c77",
         "rel": "self"
      },
      {
         "href": "http://openstack.example.com/openstack/images/c905cedb-7281-47e4-8a62-f26bc5fc4c77",
         "rel": "bookmark"
      },
      {
         "href": "http://glance.openstack.example.com/openstack/images/c905cedb-7281-47e4-8a62-f26bc5fc4c77",
         "rel": "alternate",
         "type": "application/vnd.openstack.image"
      }
   ],
   "metadata": {
      "kernel_id": "155d900f-4e14-4e4c-a73d-069cbf4541e6",
      "ramdisk_id": null
   },
   "minDisk": 0,
   "minRam": 0,
   "name": "fakeimage123456",
   "progress": 100,
   "status": "ACTIVE",
   "updated": "2011-01-01T01:02:03Z"
},
{
   "created": "2011-01-01T01:02:03Z",
   "id": "cedef40a-ed67-4d10-800e-17455edce175",
   "links": [
      
      
```
### 1.2.6.12 Get image details

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/images/{image_id}</td>
<td>Gets details for a specified image.</td>
</tr>
</tbody>
</table>

Normal response codes: 200, 203

**Request**

This table shows the URI parameters for the get image details request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{image_id}</td>
<td>UUID</td>
<td>The UUID for the image.</td>
</tr>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. Get image details: JSON response**

```json
{
    "image": {
        "created": "2011-01-01T01:02:03Z",
        "id": "70a599e0-31e7-49b7-b260-868f441e862b",
        "links": [
            {
                "href": "http://openstack.example.com/v2/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
                "rel": "self"
            },
            {
                "href": "http://openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
                "rel": "bookmark"
            },
            {
                "href": "http://glance.openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
                "rel": "alternate",
                "type": "application/vnd.openstack.image"
            }
        ],
        "metadata": {
            "architecture": "x86_64",
            "auto_disk_config": "True",
            "kernel_id": "nokernel",
            "ramdisk_id": "nokernel"
        },
        "minDisk": 0,
        "minRam": 0,
        "name": "fakeimage7",
        "progress": 100,
        "status": "ACTIVE",
        "updated": "2011-01-01T01:02:03Z"
    }
}
```
1.2.6.13 Delete image

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DELETE</td>
<td>/v2/{tenant_id}/images/{image_id}</td>
<td>Deletes a specified image.</td>
</tr>
</tbody>
</table>

Normal response codes: 204

Caution: If the password of the user who registered (imported) the image was changed, image deletion will fail. In such a case, convey the image UUID to the operator and request deletion of the image.

Request

This table shows the URI parameters for the delete image request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{image_id}</td>
<td>UUID</td>
<td>The UUID for the image.</td>
</tr>
</tbody>
</table>

This operation does not accept a request body and does not return a response body.

1.2.6.14 Show image metadata

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/images/{image_id}/metadata</td>
<td>Shows metadata for a specified image.</td>
</tr>
</tbody>
</table>

Normal response codes: 200, 203

Request

This table shows the URI parameters for the show image metadata request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{image_id}</td>
<td>UUID</td>
<td>The UUID for the image.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. Show image metadata: JSON response

```json
{
  "metadata": {
    "architecture": "x86_64",
    "auto_disk_config": "True",
    "kernel_id": "nokernel",
    "ramdisk_id": "nokernel"
  }
}
```
### 1.2.6.15 Create or replace image metadata

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUT</td>
<td>/v2/{tenant_id}/images/{image_id}/metadata</td>
<td>Creates or replaces metadata for a specified image.</td>
</tr>
</tbody>
</table>

Replaces items that match the specified keys. If you omit a key that already exists, this key retains its value.

If the number of metadata items exceeds the quota for metadata items, an overLimit (413) fault might be thrown.

Normal response codes: 200

**Caution**

Metadata cannot be changed for images that do not have a checksum set. Refer to "Update image on page 194" for details on how to change the metadata for images that do not have a checksum set.

### Request

This table shows the URI parameters for the create or replace image metadata request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{image_id}</td>
<td>UUID</td>
<td>The UUID for the image.</td>
</tr>
</tbody>
</table>

### Example. Create or replace image metadata: JSON request

```
{
  "metadata": {
    "auto_disk_config": "True",
    "Label": "Changed"
  }
}
```

### Response

### Example. Create or replace image metadata: JSON response

```
{
  "metadata": {
    "Label": "Changed",
    "auto_disk_config": "True"
  }
}
```

### 1.2.6.16 Update image metadata items

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/images/{image_id}/metadata</td>
<td>Updates the metadata items (identified by key) of the specified image.</td>
</tr>
</tbody>
</table>
Replaces items that match the specified keys and does not modify items not specified in the request.

An overLimit (413) fault might be thrown if the operation causes the quota for metadataitems to be exceeded.

Normal response codes: 200

Metadata cannot be set for images that do not have a checksum set. Refer to "Update image" on page 194 for details on how to set the metadata for images that do not have a checksum set.

### Request

This table shows the URI parameters for the update image metadata items request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{image_id}</td>
<td>UUID</td>
<td>The UUID for the image.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the update image metadata items request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>metadata</td>
<td>hash</td>
<td>Set of key/value pairs. These pairs replace any existing key/value pairs in the resources metadata with matching keys. Key/value pairs in the parameter with keys that do not occur in the existing resource metadata are added to the resources metadata.</td>
</tr>
</tbody>
</table>

**Example. Update image metadata items: JSON request**

```
{
    "metadata": {
        "kernel_id": "False",
        "Label": "UpdatedImage"
    }
}
```

**Response**

**Example. Update image metadata items: JSON response**

```
{
    "metadata": {
        "Label": "UpdatedImage",
        "architecture": "x86 64",
        "auto_disk_config": "True",
        "kernel_id": "False",
        "ramdisk_id": "nokernel"
    }
}
```
1.2.6.17 Show image metadata item details

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/images/{image_id}/metadata/{key}</td>
<td>Shows details for a metadata item by key for a specified image.</td>
</tr>
</tbody>
</table>

Normal response codes: 200, 203

**Request**

This table shows the URI parameters for the show image metadata item details request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{image_id}</td>
<td>UUID</td>
<td>The UUID for the image.</td>
</tr>
<tr>
<td>{key}</td>
<td>ImageMetadataKey</td>
<td>A string. Maximum length is 255 characters.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. Show image metadata item details: JSON response**

```json
{
   "meta": {
      "architecture": "x86_64",
      "auto_disk_config": "True",
      "kernel_id": "nokernel",
      "ramdisk_id": "nokernel"
   }
}
```

1.2.6.18 Create or update image metadata item

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUT</td>
<td>/v2/{tenant_id}/images/{image_id}/metadata/{key}</td>
<td>Creates or updates the metadata items (identified by key) of the specified image.</td>
</tr>
</tbody>
</table>

An overLimit (413) fault might be thrown if the operation causes the quota for metadata items to be exceeded.

Normal response codes: 200

⚠️ **Caution** Metadata cannot be changed for images that do not have a checksum set. Refer to "Update image" on page 194 for details on how to change the metadata for images that do not have a checksum set.
Request

This table shows the URI parameters for the create or update image metadata item:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{image_id}</td>
<td>UUID</td>
<td>The UUID for the image.</td>
</tr>
<tr>
<td>{key}</td>
<td>Image</td>
<td>Metadata Key</td>
</tr>
<tr>
<td></td>
<td>Metadata</td>
<td>Opcion.</td>
</tr>
<tr>
<td></td>
<td>Key</td>
<td>A string. Maximum length is 255 characters.</td>
</tr>
</tbody>
</table>

Example. Create or update image metadata item: JSON request

```json
{
    "meta": {
        "auto_disk_config": "True",
        "Label": "Changed"
    }
}
```

Response

Example. Create or update image metadata item: JSON response

```json
{
    "meta": {
        "Label": "Changed",
        "auto_disk_config": "True"
    }
}
```

1.2.6.19 Delete image metadata item

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DELETE</td>
<td>/v2/{tenant_id}/images/{image_id}/metadata/{key}</td>
<td>Deletes a metadata item by key for a specified image.</td>
</tr>
</tbody>
</table>

Normal response codes: 204

Request

This table shows the URI parameters for the delete image metadata item request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{image_id}</td>
<td>UUID</td>
<td>The UUID for the image.</td>
</tr>
<tr>
<td>{key}</td>
<td>Image</td>
<td>Metadata Key</td>
</tr>
<tr>
<td></td>
<td>Metadata</td>
<td>Opcion.</td>
</tr>
<tr>
<td></td>
<td>Key</td>
<td>A string. Maximum length is 255 characters.</td>
</tr>
</tbody>
</table>
This operation does not accept a request body and does not return a response body.

### 1.2.6.20 Create server with scheduler hints

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers</td>
<td>Creates a server with scheduler hints that are passed directly to the scheduler.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

**Caution**

- When an instance created using a Linux image is started or restarted, the sshd settings are initialized, and it may no longer be possible to log in to the instance. Refer to "[Reboot server](#) on page 26" for details on the required tasks.
- When the password of the user who registered (imported) the image is changed, the image can no longer be used. In such a case, convey either of the following to the operator and request a change of image.
  - The location of the image following the change of image UUID and user authentication information
  - The image UUID that was registered (imported) following the change of image UUID and user authentication information

**Request**

This table shows the URI parameters for the create server with scheduler hints request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the create server with scheduler hints request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>security_group</td>
<td>string (Optional)</td>
<td>One or more security_group objects. Specify the name of the security group in the name attribute. If you omit this attribute, the server is created in the default security group. Specify a security group for which TCP communication is permitted for the following IP address and port number. IP address: 169.254.169.254 Port number: 80 If TCP communication is not permitted, the host name (computer name) and administrator password may not be set when creating an instance.</td>
</tr>
<tr>
<td>user_data</td>
<td>string (Optional)</td>
<td>Configuration information or scripts to use upon launch. Must be Base64 encoded. The main formats that are supported are as follows:</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Linux:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Shell script (begins with <code>#!</code>)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Windows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PowerShell (begins with <code>#ps1_sysnative</code> or <code>#ps1_x86</code>)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Windows batch (begins with <code>rem cmd</code>)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If Linux, cloud-config files can also be specified, but since verification has not been completed, we recommend using Shell scripts.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>availability_zone</th>
<th>string (Optional)</th>
<th>The availability zone in which to launch the server.</th>
</tr>
</thead>
<tbody>
<tr>
<td>server</td>
<td>ServerForCreate</td>
<td>server.</td>
</tr>
<tr>
<td>imageRef</td>
<td>string</td>
<td>The image reference for the desired image for your server instance. Specify as an ID or full URL.</td>
</tr>
<tr>
<td>flavorRef</td>
<td>string</td>
<td>The flavor reference for the desired flavor for your server instance. Specify as an ID or full URL.</td>
</tr>
<tr>
<td>key_name</td>
<td>string (Optional)</td>
<td>Assigns the public key of the named keypair to the server.</td>
</tr>
<tr>
<td>networks</td>
<td>string (Optional)</td>
<td>A networks object. By default, the server instance is provisioned with all isolated networks for the tenant.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Optionally, you can create one or more NICs on the server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To provision the server instance with a NIC for a network, specify the UUID of the network in the uuid attribute in a networks object.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To provision the server instance with a NIC for an already existing port, specify the port-id in the port attribute in a networks object.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You can specify multiple NICs on the server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ensure that the network is specified.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To set the following information for a virtual server, a virtual router must be connected to the network that the virtual server will be connected to.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Host name (computer name)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Administrator password</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Authentication key (key pair)</td>
</tr>
<tr>
<td>uuid</td>
<td>string (Optional)</td>
<td>To provision the server instance with a NIC for a network, specify the UUID of the network in the uuid attribute in a networks object. Required if you omit the port attribute.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>if the network includes multiple subnets, IP addresses will be allocated from any subnet. To allocate an IP address of a specific subnet, create a port in advance, and specify the uuid of that port in port.</td>
<td>port</td>
<td>string (Optional) To provision the server instance with a NIC for an already existing port, specify the port-id in the port attribute in a networks object. Required if you omit the uuid attribute.</td>
</tr>
<tr>
<td>A fixed IPv4 address for the NIC. Valid with a neutron or nova-networks network.</td>
<td>fixed_ip</td>
<td>string (Optional)</td>
</tr>
<tr>
<td>The server name. This information is also used as the computer name/host name. If 64 characters or more are specified: • Windows: The computer name is the default name set by Windows. • Linux: The host name will be &quot;host-fixedIpAddressOfEth0&quot;. The string set for the computer name/host name is changed as follows: • Spaces ( ) and underscores (_) are replaced with hyphens (-) • Uppercase letters are replaced with lowercase letters • Symbols other than periods (.) and hyphens (-) are removed • Periods (.) are removed from the beginning and end of the string, and strings consisting of hyphens (-) are removed • In Windows, if the string contains a period (.) anywhere other than at the beginning or end, then the characters preceding the period (.) will be used for the name.</td>
<td>name</td>
<td>string (Optional)</td>
</tr>
<tr>
<td>Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each. Windows: To specify a password of an instance, specify the &quot;admin_pass&quot; key. The specified password will be set for users specified in cloudbase-init. Check with the image provider regarding users who are specified for cloudbase-init. Example setting: &quot;metadata&quot;: {&quot;admin_pass&quot;: &quot;passwordSetForInstance&quot;}</td>
<td>metadata</td>
<td>string (Optional)</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>block_device_mapping_v2</td>
<td>string</td>
<td>Enables booting the server from a volume when additional parameters are given.</td>
</tr>
<tr>
<td>device_name</td>
<td>string</td>
<td>Describes a path to the device for the volume you want to use to boot the server. Specify this item in /dev/vd/deviceName format. /dev/vd is fixed, and for deviceName, specify characters that are valid as a device name. When creating an instance that is allocated multiple volumes, for the boot volume, specify the character with highest priority among the device names of all volumes. The order of priority is a &gt; b &gt; c &gt; ...</td>
</tr>
<tr>
<td>source_type</td>
<td>string</td>
<td>Describes the volume source type for the volume. Choices are &quot;snapshot&quot;, &quot;volume&quot;, or &quot;image&quot;.</td>
</tr>
<tr>
<td>destination_type</td>
<td>string</td>
<td>Specifies the connection destination (&quot;volume&quot;).</td>
</tr>
<tr>
<td>delete_on_termination</td>
<td>bool</td>
<td>Specifies whether volumes created during the instance creation will be deleted when the instance is deleted. When &quot;True&quot; is specified, volumes during instance creation will also be deleted when the instance is deleted. When &quot;False&quot; is specified, the volumes created during instance creation will not be deleted when the instance is deleted. If not specified, &quot;False&quot; (do not delete) will be used. The volume where snapshots are collected will not be deleted even if &quot;True&quot; is specified.</td>
</tr>
<tr>
<td>boot_index</td>
<td>string</td>
<td>Specifies the device start order. Specify sequential values, starting from 0. For the boot disk, specify &quot;0&quot;.</td>
</tr>
<tr>
<td>config_drive</td>
<td>string</td>
<td>Only &quot;false&quot; can be specified.</td>
</tr>
<tr>
<td>uuid</td>
<td>uuid</td>
<td>Specifies the UUID of the resource specified for source_type.</td>
</tr>
<tr>
<td>volume_size</td>
<td>string</td>
<td>Specify the volume size in GB. This item must be specified when &quot;image&quot; is specified for source_type. Specify a value equal to or higher than the min_disk parameter of the image to be used. If the min_disk parameter of the image to be used has not been specified or is &quot;0&quot;, check the minimum size with the image provider and specify the value accordingly. If &quot;volume&quot; was specified for source_type, this item will be ignored even if a value is specified.</td>
</tr>
</tbody>
</table>
If "snapshot" was specified for source_type, and this item is omitted, the volume size of the snapshot collection source will be used.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>os:scheduler_hints</td>
<td>dict</td>
<td>Dictionary of data to send to the scheduler.</td>
</tr>
</tbody>
</table>

When trying to create an instance with the server group UUID of "anti-affinity" but there is no VM host where it can be created, its status becomes ERROR after the creation request is received.

If using the dedicated instance feature, specify "fcx.dedicated": "true".

This will be ignored if the request body does not specify the correct hierarchy (that is, the same hierarchy as the server).

Example. Create server with scheduler hints: JSON request

```json
{
    "server": {
        "name": "new-server-test",
        "imageRef": "70a599e0-31e7-49b7-b260-868f441e862b",
        "flavorRef": "1",
        "key_name": "keypair1"
    },
    "os:scheduler_hints": {
        "group": "2b7c42eb-7736-4a0f-afab-f23969a35ada"
    }
}
```

Response

Example. Create server with scheduler hints: JSON response

```json
{
    "server": {
        "adminPass": "yjzytFHb7XHc",
        "id": "f8f4f3ce-f6e0-4e05-8f79-bf984fdfe45",
        "links": [
            {
                "href": "http://openstack.example.com/v2/openstack/servers/f8f4f3ce-f6e0-4e05-8f79-bf984fdfe45",
                "rel": "self"
            },
            {
                "href": "http://openstack.example.com/openstack/servers/f8f4f3ce-f6e0-4e05-8f79-bf984fdfe45",
                "rel": "bookmark"
            }
        ]
    }
}
```
1.2.6.21 Create multiple servers

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers</td>
<td>Creates one or more servers with an optional reservation ID.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

Caution

- When an instance created using a Linux image is started or restarted, the sshd settings are initialized, and it may no longer be possible to log in to the instance. Refer to "Reboot server" on page 26 for details on the required tasks.
- When the password of the user who registered (imported) the image is changed, the image can no longer be used. In such a case, convey either of the following to the operator and request a change of image.
  - The location of the image following the change of image UUID and user authentication information
  - The image UUID that was registered (imported) following the change of image UUID and user authentication information

Request

This table shows the URI parameters for the create multiple servers request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This table shows the request body parameters for the create multiple servers request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>security_group</td>
<td>string (Optional)</td>
<td>One or more security_group objects. Specify the name of the security group in the name attribute. If you omit this attribute, the server is created in the default security group. Specify a security group for which TCP communication is permitted for the following IP address and port number. IP address: 169.254.169.254 Port number: 80 If TCP communication is not permitted, the host name (computer name) and administrator password may not be set when creating an instance.</td>
</tr>
</tbody>
</table>
| user_data     | string (Optional) | Configuration information or scripts to use upon launch. Must be Base64 encoded. The main formats that are supported are as follows:
  - Linux:
  - Shell script (begins with #!) |
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>availability_zone</td>
<td>string (Optional)</td>
<td>The availability zone in which to launch the server.</td>
</tr>
<tr>
<td>server</td>
<td>ServerForCreate</td>
<td>server.</td>
</tr>
<tr>
<td>imageRef</td>
<td>string</td>
<td>The image reference for the desired image for your server instance. Specify as an ID or full URL.</td>
</tr>
<tr>
<td>flavorRef</td>
<td>string</td>
<td>The flavor reference for the desired flavor for your server instance. Specify as an ID or full URL.</td>
</tr>
<tr>
<td>key_name</td>
<td>string (Optional)</td>
<td>Assigns the public key of the named keypair to the server.</td>
</tr>
<tr>
<td>networks</td>
<td>string (Optional)</td>
<td>A networks object. By default, the server instance is provisioned with all isolated networks for the tenant. Optionally, you can create one or more NICs on the server. To provision the server instance with a NIC for a network, specify the UUID of the network in the uuid attribute in a networks object. To set the following information for a virtual server, a virtual router must be connected to the network that the virtual server will be connected to. • Host name (computer name) • Administrator password • Authentication key (key pair)</td>
</tr>
<tr>
<td>uuid</td>
<td>string (Optional)</td>
<td>To provision the server instance with a NIC for a network, specify the UUID of the network in the uuid attribute in a networks object. Required if you omit the port attribute. If the network includes multiple subnets, IP addresses will be allocated from any subnet. To allocate an IP address of a specific subnet, create a port in advance, and specify the uuid of that port in port.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>port</td>
<td>string</td>
<td>To provision the server instance with a NIC for an already existing port, specify the port-id in the port attribute in a networks object. Required if you omit the uuid attribute.</td>
</tr>
<tr>
<td>fixed_ip</td>
<td>string</td>
<td>A fixed IPv4 address for the NIC. Valid with a neutron or nova-networks network.</td>
</tr>
<tr>
<td>name</td>
<td>string</td>
<td>The server name. This information is also used as the computer name/host name. If 64 characters or more are specified:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Windows: The computer name will be the default name set by Windows.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Linux: The host name will be &quot;host-fixedIpAddressOfEth0&quot;.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The string set for the computer name/host name is changed as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Spaces ( ) and underscores (_) are replaced with hyphens (-)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Uppercase letters are replaced with lowercase letters</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Symbols other than periods (.) and hyphens (-) are removed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Periods (.) are removed from the beginning and end of the string, and strings consisting of hyphens (-) are removed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• In Windows, if the string contains a period (.) anywhere other than at the beginning or end, then the characters preceding the period (.) will be used for the name.</td>
</tr>
<tr>
<td>metadata</td>
<td>string</td>
<td>Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows: To specify a password of an instance, specify the &quot;admin_pass&quot; key.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The specified password will be set for users specified in cloudbase-init.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Check with the image provider regarding users who are specified for cloudbase-init.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example setting: &quot;metadata&quot;: {&quot;admin_pass&quot;: &quot;passwordSetForInstance&quot;}</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To use the automatic failover feature, specify &quot;fcx.autofailover&quot;: &quot;true&quot;.</td>
</tr>
<tr>
<td>block_device_mapping_v2</td>
<td>string</td>
<td>Enables booting the server from a volume when additional parameters are given.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>device_name</td>
<td>string</td>
<td>Describes a path to the device for the volume you want to use to boot the server. Specify this item in <code>/dev/vddeviceName</code> format. <code>/dev/vd</code> is fixed, and for <code>deviceName</code>, specify characters that are valid as a device name. When creating an instance that is allocated multiple volumes, for the boot volume, specify the character with highest priority among the device names of all volumes. The order of priority is <code>a &gt; b &gt; c &gt; ...</code></td>
</tr>
<tr>
<td>source_type</td>
<td>string</td>
<td>Describes the volume source type for the volume. Choices are &quot;snapshot&quot;, &quot;volume&quot;, or &quot;image&quot;.</td>
</tr>
<tr>
<td>destination_type</td>
<td>string</td>
<td>Specifies the connection destination (&quot;volume&quot;).</td>
</tr>
<tr>
<td>delete_on_termination</td>
<td>bool</td>
<td>Specifies whether volumes created during the instance creation will be deleted when the instance is deleted. When &quot;True&quot; is specified, volumes during instance creation will also be deleted when the instance is deleted. When &quot;False&quot; is specified, the volumes created during instance creation will not be deleted when the instance is deleted. If not specified, &quot;False&quot; (do not delete) will be used. The volume where snapshots are collected will not be deleted even if &quot;True&quot; is specified.</td>
</tr>
<tr>
<td>boot_index</td>
<td>String</td>
<td>Specifies the device start order. Specify sequential values, starting from 0. For the boot disk, specify &quot;0&quot;.</td>
</tr>
<tr>
<td>config_drive</td>
<td>String</td>
<td>Only &quot;false&quot; can be specified.</td>
</tr>
<tr>
<td>uuid</td>
<td>uuid</td>
<td>Specifies the UUID of the resource specified for source_type.</td>
</tr>
<tr>
<td>volume_size</td>
<td>String</td>
<td>Specify the volume size in GB. This item must be specified when &quot;image&quot; is specified for source_type. Specify a value equal to or higher than the min_disk parameter of the image to be used. If the min_disk parameter of the image to be used has not been specified or is &quot;0&quot;, check the minimum size with the image provider and specify the value accordingly. If &quot;volume&quot; was specified for source_type, this item will be ignored even if a value is specified. If &quot;snapshot&quot; was specified for source_type, and this item is omitted, the volume size of the snapshot collection source will be used.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>--------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>return_reservation_id</td>
<td>string (Optional)</td>
<td>Set to True to generate a reservation ID for each server. Omit this attribute to create servers without a reservation ID. This extended attribute is enabled when the service provider enables multiple server launch.</td>
</tr>
<tr>
<td>min_count</td>
<td>string (Optional)</td>
<td>The minimum number of servers to launch when the service provider enables multiple server launch. An error will occur if unable to create the specified minimum number of servers. If this item is omitted, the 1 will be used.</td>
</tr>
<tr>
<td>max_count</td>
<td>string (Optional)</td>
<td>The maximum number of servers to launch when the service provider enables multiple server launch. The system will try to create the specified number of servers. If this item is omitted, min_count will be used.</td>
</tr>
</tbody>
</table>

**Example. Create multiple servers: JSON request**

Creates one or more servers with an optional reservation ID.

```json
{
    "server": {
        "name": "new-server-test",
        "imageRef": "9f033140-ea8c-41fe-a432-e832799aa47f",
        "flavorRef": "1",
        "key_name": "keypair1",
        "metadata": {
            "My Server Name": "Apache1"
        },
        "return_reservation_id": true
    }
}
```

Creates one or more servers with an optional min count.

```json
{
    "server": {
        "name": "new-server-test",
        "imageRef": "9f033140-ea8c-41fe-a432-e832799aa47f",
        "flavorRef": "08ef25d1-9616-46b6-bad3-3835efccf3a5",
        "metadata": {
            "My Server Name": "Apache1"
        },
        "min_count": 1
    }
}
```

Creates one or more servers with an optional max count.

```json
{
    "server": {
        "name": "new-server-test",
        "imageRef": "9f033140-ea8c-41fe-a432-e832799aa47f",
        "flavorRef": "08ef25d1-9616-46b6-bad3-3835efccf3a5",
        "metadata": {
            "My Server Name": "Apache1"
        },
        "max_count": 1
    }
}
```
Response

Example. Create multiple servers: JSON response

```json
{
  "server": {
    "adminPass": "wfksH3GTTseP",
    "id": "440cf918-3ee0-4143-b289-f63e1d2000e6",
    "links": [
      {
        "href": "http://openstack.example.com/v2/openstack/
          servers/440cf918-3ee0-4143-b289-f63e1d2000e6",
        "rel": "self"
      },
      {
        "href": "http://openstack.example.com/openstack/
          servers/440cf918-3ee0-4143-b289-f63e1d2000e6",
        "rel": "bookmark"
      }
    ]
  }
}
```

Example. Create multiple servers: JSON response

```json
{
  "reservation_id": "r-3fhpjulh"
}
```

1.2.6.22 Show server extended status

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/{server_id}</td>
<td>Shows the extended status attributes in the response for a specified server.</td>
</tr>
</tbody>
</table>

The extended status attributes are vm_state, power_state, and task_state.

Normal response codes: 200

Request

This table shows the URI parameters for the show server extended status request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.
Example. Show server extended status: JSON response

```json
{
    "server": {
        "OS-EXT-STS:power_state": 1,
        "OS-EXT-STS:task_state": null,
        "OS-EXT-STS:vm_state": "active",
        "accessIPv4": "",
        "accessIPv6": "",
        "addresses": {
            "private": [
                {
                    "addr": "192.168.0.3",
                    "version": 4
                }
            ]
        },
        "created": "2013-02-07T19:35:09Z",
        "flavor": {
            "id": "1",
            "links": [
                {
                    "href": "http://openstack.example.com/openstack/flavors/1",
                    "rel": "bookmark"
                }
            ]
        },
        "hostId": "570eff4776ab310707d11d181037337197086998a8b3305c90bf87c8",
        "id": "ecb5e433-fa75-4db2-af3d-a29ae8618edc",
        "image": {
            "id": "70a599e0-31e7-49b7-b260-868f441e862b",
            "links": [
                {
                    "href": "http://openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
                    "rel": "bookmark"
                }
            ]
        },
        "links": [
            {
                "href": "http://openstack.example.com/v2/openstack/servers/ecb5e433-fa75-4db2-af3d-a29ae8618edc",
                "rel": "self"
            },
            {
                "href": "http://openstack.example.com/openstack/servers/ecb5e433-fa75-4db2-af3d-a29ae8618edc",
                "rel": "bookmark"
            }
        ],
        "metadata": {
            "My Server Name": "Apache1"
        },
        "name": "new-server-test",
        "progress": 0,
        "status": "ACTIVE",
        "tenant_id": "openstack",
        "updated": "2013-02-07T19:35:10Z",
        "user_id": "fake"
    }
}
```
### 1.2.6.23 List extended status for servers

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/detail</td>
<td>Lists the extended status attributes in the detailed response for all servers.</td>
</tr>
</tbody>
</table>

The extended status attributes are `vm_state`, `power_state`, and `task_state`.

Normal response codes: 200

### Request

This table shows the URI parameters for the list extended status for servers request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

### Response

**Example. List extended status for servers: JSON response**

```json
{
  "servers": [
    {
      "OS-EXT-STS:power_state": 1,
      "OS-EXT-STS:task_state": null,
      "OS-EXT-STS:vm_state": "active",
      "accessIPv4": "",
      "accessIPv6": "",
      "addresses": {
        "private": [
          {
            "addr": "192.168.0.3",
            "version": 4
          }
        ]
      },
      "created": "2012-12-05T07:34:10Z",
      "flavor": {
        "id": "1",
        "links": [
          {
            "href": "http://openstack.example.com/openstack/flavors/1",
            "rel": "bookmark"
          }
        ]
      },
      "hostId": "585aa01f94eca692eff9f77ffe3eab866d8a819e97397e28c5c7df12",
      "id": "030758aa-5c41-41c6-8fb4-29d44eb96a85",
      "image": {
        "id": "70a599e0-31e7-49b7-b260-868f441e862b",
        "links": [
          {
            "href": "http://openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
            "rel": "bookmark"
          }
        ]
      }
    }
  ]
}```
1.2.6.24 List servers with extended server attributes

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers</td>
<td>Lists detailed extended server attribute information for all servers.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. List servers with extended server attributes: JSON response**

```json
{
    "servers": [
        {
            "id": "616fb98f-46ca-475e-917e-2563e5a8cd19",
            "links": [
                {
                    "href": "http://openstack.example.com/v2/openstack/servers/616fb98f-46ca-475e-917e-2563e5a8cd19",
                    "rel": "self"
                }
            ]
        }
    ]
}
```
1.2.6.25 Show extended server attributes

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/{server_id}</td>
<td>Shows extended server attributes for a specified server.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the show extended server attributes request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. Show extended server attributes: JSON response**

```json
{
  "server": {
    "OS-EXT-SRV-ATTR:host": "1169a68456af49238da47b1d5957a714",
    "OS-EXT-SRV-ATTR:hostname": "fake-mini",
    "OS-EXT-SRV-ATTR:instance_name": "instance-00000001",
    "accessIPv4": "",
    "accessIPv6": "",
    "addresses": {
      "private": [
        { "addr": "192.168.0.3" },
```
1.2.6.26 List servers with IP type

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/detail</td>
<td>Lists all servers showing their IPs by type, either fixed or floating.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the list servers with ip type request:
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. List servers with IP type: JSON response**

```json
{
    "servers": [
        {
            "accessIPv4": "",
            "accessIPv6": "",
            "addresses": {
                "private": [
                    {
                        "OS-EXT-IPS:type": "fixed",
                        "addr": "192.168.0.3",
                        "version": 4
                    }
                ]
            },
            "created": "2013-02-07T18:40:59Z",
            "flavor": {
                "id": "1",
                "links": [
                    {
                        "href": "http://openstack.example.com/openstack/flavors/1",
                        "rel": "bookmark"
                    }
                ]
            },
            "hostId": "fe866a4962fe3dbb6c2db9c8f7dcdcb9555aca73387e72b5cb9c45bd3",
            "id": "76908712-653a-4d16-807e-d89d41435d24",
            "image": {
                "id": "70a599e0-31e7-49b7-b260-868f441e862b",
                "links": [
                    {
                        "href": "http://openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
                        "rel": "bookmark"
                    }
                ]
            },
            "links": [
                {
                    "href": "http://openstack.example.com/v2/openstack/servers/76908712-653a-4d16-807e-d89d41435d24",
                    "rel": "self"
                },
                {
                    "href": "http://openstack.example.com/openstack/servers/76908712-653a-4d16-807e-d89d41435d24",
                    "rel": "bookmark"
                }
            ],
            "metadata": {
                "My Server Name": "Apache1"
            },
            "name": "new-server-test",
            "progress": 0,
            "status": "ACTIVE",
```
1.2.6.27 Create server

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers</td>
<td>Creates a server.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

Caution

- When an instance created using a Linux image is started or restarted, the sshd settings are initialized, and it may no longer be possible to log in to the instance. Refer to "Reboot server" on page 26 for details on the required tasks.
- When the password of the user who registered (imported) the image is changed, the image can no longer be used. In such a case, convey either of the following to the operator and request a change of image.
  - The location of the image following the change of image UUID and user authentication information
  - The image UUID that was registered (imported) following the change of image UUID and user authentication information

Request

This table shows the URI parameters for the create server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the create server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>security_group</td>
<td>string (Optional)</td>
<td>One or more security_group objects. Specify the name of the security group in the name attribute. If you omit this attribute, the server is created in the default security group. Specify a security group for which TCP communication is permitted for the following IP address and port number. IP address: 169.254,169.254 Port number: 80 If TCP communication is not permitted, the host name (computer name) and administrator password may not be set when creating an instance.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>user_data</td>
<td>string</td>
<td>Configuration information or scripts to use upon launch. Must be Base64 encoded. The main formats that are supported are as follows:</td>
</tr>
<tr>
<td></td>
<td>(Optional)</td>
<td>• Linux:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Shell script (begins with <code>#!</code>)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Windows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PowerShell (begins with <code>#ps1_sysnative</code> or <code>#ps1_x86</code>)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Windows batch (begins with <code>rem cmd</code>)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If Linux, cloud-config files can also be specified, but since verification has not been completed, we recommend using Shell scripts.</td>
</tr>
<tr>
<td>availability_zone</td>
<td>string</td>
<td>The availability zone in which to launch the server.</td>
</tr>
<tr>
<td>server</td>
<td>ServerForCreate</td>
<td>server.</td>
</tr>
<tr>
<td>imageRef</td>
<td>string</td>
<td>The image reference for the desired image for your server instance. Specify as an ID or full URL.</td>
</tr>
<tr>
<td>flavorRef</td>
<td>string</td>
<td>The flavor reference for the desired flavor for your server instance. Specify as an ID or full URL.</td>
</tr>
<tr>
<td>key_name</td>
<td>string</td>
<td>Assigns the public key of the named keypair to the server.</td>
</tr>
<tr>
<td>key_name</td>
<td>(Optional)</td>
<td></td>
</tr>
<tr>
<td>networks</td>
<td>string</td>
<td>A networks object. By default, the server instance is provisioned with all isolated networks for the tenant.</td>
</tr>
<tr>
<td></td>
<td>(Optional)</td>
<td>Optionally, you can create one or more NICs on the server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To provision the server instance with a NIC for a network, specify the UUID of the network in the uuid attribute in a networks object.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To provision the server instance with a NIC for an already existing port, specify the port-id in the port attribute in a networks object.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You can specify multiple NICs on the server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ensure that the network is specified.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To set the following information for a virtual server, a virtual router must be connected to the network that the virtual server will be connected to.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Host name (computer name)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Administrator password</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Authentication key (key pair)</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>---------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| uuid     | string (Optional) | To provision the server instance with a NIC for a network, specify the UUID of the network in the uuid attribute in a networks object. Required if you omit the port attribute.  
If the network includes multiple subnets, IP addresses will be allocated from any subnet. To allocate an IP address of a specific subnet, create a port in advance, and specify the uuid of that port in port. |
| port     | string (Optional) | To provision the server instance with a NIC for an already existing port, specify the port-id in the port attribute in a networks object. Required if you omit the uuid attribute. |
| fixed_ip | string (Optional) | A fixed IPv4 address for the NIC. Valid with a neutron or nova-networks network.                                                             |
| name     | string        | The server name. This information is also used as the computer name/host name.  
If 64 characters or more are specified:  
• Windows:  
The computer name is the default name set by Windows.  
• Linux:  
The host name will be "host-fixedIpAddressOfEth0".  
The string set for the computer name/host name is changed as follows:  
• Spaces ( ) and underscores (_) are replaced with hyphens (-)  
• Uppercase letters are replaced with lowercase letters  
• Symbols other than periods (.) and hyphens (-) are removed  
• Periods (.) are removed from the beginning and end of the string, and strings consisting of hyphens (-) are removed  
• In Windows, if the string contains a period (.) anywhere other than at the beginning or end, then the characters preceding the period (.) will be used for the name. |
| metadata | string (Optional) | Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each.  
Windows:  
To specify a password of an instance, specify the "admin_pass" key.  
The specified password will be set for users specified in cloudbase-init. |
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Check with the image provider regarding users who are specified for cloudbase-init. Example setting: <em>metadata</em>: {&quot;admin_pass&quot;: &quot;passwordSetForInstance&quot;} To use the automatic failover feature, specify &quot;fcx.autofailover&quot;: &quot;true&quot;.</td>
</tr>
<tr>
<td>block_device_mapping_v2</td>
<td>string</td>
<td>Enables booting the server from a volume when additional parameters are given.</td>
</tr>
<tr>
<td>device_name</td>
<td>string</td>
<td>Describes a path to the device for the volume you want to use to boot the server. Specify this item in /dev/vd&lt;deviceName&gt; format. /dev/vd is fixed, and for &lt;deviceName&gt;, specify characters that are valid as a device name. When creating an instance that is allocated multiple volumes, for the boot volume, specify the character with highest priority among the device names of all volumes. The order of priority is a &gt; b &gt; c &gt; ...</td>
</tr>
<tr>
<td>source_type</td>
<td>string</td>
<td>Describes the volume source type for the volume. Choices are &quot;snapshot&quot;, &quot;volume&quot;, or &quot;image&quot;.</td>
</tr>
<tr>
<td>destination_type</td>
<td>string</td>
<td>Specifies the connection destination (&quot;volume&quot;).</td>
</tr>
<tr>
<td>delete_on_termination</td>
<td>bool</td>
<td>Specifies whether volumes created during the instance creation will be deleted when the instance is deleted. When &quot;True&quot; is specified, volumes during instance creation will also be deleted when the instance is deleted. When &quot;False&quot; is specified, the volumes created during instance creation will not be deleted when the instance is deleted. If not specified, &quot;False&quot; (do not delete) will be used. The volume where snapshots are collected will not be deleted even if &quot;True&quot; is specified.</td>
</tr>
<tr>
<td>boot_index</td>
<td>string</td>
<td>Specifies the device start order. Specify sequential values, starting from 0. For the boot disk, specify &quot;0&quot;.</td>
</tr>
<tr>
<td>config_drive</td>
<td>String</td>
<td>Only &quot;false&quot; can be specified.</td>
</tr>
<tr>
<td>uuid</td>
<td>uuid</td>
<td>Specifies the UUID of the resource specified for source_type.</td>
</tr>
<tr>
<td>volume_size</td>
<td>string</td>
<td>Specify the volume size in GB. This item must be specified when &quot;image&quot; is specified for source_type. Specify a value equal to or higher than the min_disk parameter of the image to be used. If the min_disk parameter of the image to be used has not been specified or</td>
</tr>
</tbody>
</table>
is "0", check the minimum size with the image provider and specify the value accordingly.
If "volume" was specified for source_type, this item will be ignored even if a value is specified.
If "snapshot" was specified for source_type, and this item is omitted, the volume size of the snapshot collection source will be used.

Example. Create server: JSON request

```json
{
  "server": {
      "name": "new-server-test",
      "imageRef": "b5660a6e-4b46-4be3-9707-6b47221b454f",
      "flavorRef": "2",
      "key_name": "keypair1",
      "metadata": {
        "My Server Name": "Apache1"
      }
  }
}
```

Response

Example. Create server: JSON response

```json
{
  "server": {
      "OS-DCF:diskConfig": "AUTO",
      "adminPass": "CQH9gWzgkVno",
      "id": "324dfb7d-f4a9-419a-9a19-237df04b443b",
      "links": [
        {
          "href": "http://openstack.example.com/v2/openstack/servers/324dfb7d-f4a9-419a-9a19-237df04b443b",
          "rel": "self"
        },
        {
          "href": "http://openstack.example.com/openstack/servers/324dfb7d-f4a9-419a-9a19-237df04b443b",
          "rel": "bookmark"
        }
      ]
  }
}
```

1.2.6.28 Show server information

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/{server_id}</td>
<td>Shows information for a specified server.</td>
</tr>
</tbody>
</table>

Normal response codes: 200
Request

This table shows the URI parameters for the show server information request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. Show server information: JSON response

```
{
  "server": {
    "OS-DCF:diskConfig": "AUTO",
    "accessIPv4": "",
    "accessIPv6": "",
    "addresses": {
      "private": [
        { "addr": "192.168.0.3", "version": 4 }
      ]
    },
    "created": "2012-12-02T02:11:55Z",
    "flavor": {
      "id": "1",
      "links": [
        { "href": "http://openstack.example.com/openstack/flavors/1", "rel": "bookmark" }
      ]
    },
    "hostId": "c949ab4256cea23b6089b710aa2df48bf6577ed915278b62e33ad8bb",
    "id": "5046e2f2-3b33-4041-b3cf-e085f73e78e7",
    "image": {
      "id": "70a599e0-31e7-49b7-b260-868f441e862b",
      "links": [
        { "href": "http://openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b", "rel": "bookmark" }
      ]
    },
    "links": [
      { "href": "http://openstack.example.com/v2/openstack/servers/5046e2f2-3b33-4041-b3cf-e085f73e78e7", "rel": "self" },
      { "href": "http://openstack.example.com/openstack/servers/5046e2f2-3b33-4041-b3cf-e085f73e78e7", "rel": "bookmark" }
    ],
    "metadata": {
      "My Server Name": "Apache1"
    }
  }
}
```


### 1.2.6.29 List servers

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/detail</td>
<td>Lists servers.</td>
</tr>
</tbody>
</table>

Normal response codes: 200, 203

**Request**

This table shows the URI parameters for the list servers request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. List servers: JSON response**

```json
{
  "servers": [
  {
    "OS-DCF:diskConfig": "AUTO",
    "accessIPv4": "",
    "accessIPv6": "",
    "addresses": {
      "private": [
        {
          "addr": "192.168.0.3",
          "version": 4
        }
      ]
    },
    "created": "2012-12-02T02:11:55Z",
    "flavor": {
      "id": "1",
      "links": [
        {
          "href": "http://openstack.example.com/openstack/flavors/1",
          "rel": "bookmark"
        }
      ]
    },
    "hostId": "99428f33351a5d89d0f7727c66ec68c1777c545a0972aaac645508dc",
    "id": "05372e62-05b9-4ee2-9343-9a1f6f2a5fda",
```
1.2.6.30 Get image information

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/images/{image_id}</td>
<td>Gets information for a specified image.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the get image information request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{image_id}</td>
<td>String</td>
<td>The UUID for the image of interest to you.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. Get image information: JSON response**

```json
{
    "image": {
        "id": "70a599e0-31e7-49b7-b260-868f441e862b",
        "links": [
            {
                "href": "http://openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
                "rel": "bookmark"
            }
        ],
        "metadata": {
            "My Server Name": "Apache1"
        },
        "name": "new-server-test",
        "progress": 0,
        "status": "ACTIVE",
        "tenant_id": "openstack",
        "updated": "2012-12-02T02:11:56Z",
        "user_id": "fake"
    }
}
```
"OS-DCF:diskConfig": "AUTO",
"created": "2011-01-01T01:02:03Z",
"id": "70a599e0-31e7-49b7-b260-868f441e862b",
"links": [
  {
    "href": "http://openstack.example.com/v2/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
    "rel": "self"
  },
  {
    "href": "http://openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
    "rel": "bookmark"
  },
  {
    "href": "http://glance.openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
    "rel": "alternate",
    "type": "application/vnd.openstack.image"
  }
],
"metadata": {
  "architecture": "x86_64",
  "auto_disk_config": "True",
  "kernel_id": "nokernel",
  "ramdisk_id": "nokernel"
},
"minDisk": 0,
"minRam": 0,
"name": "fakeimage7",
"progress": 100,
"status": "ACTIVE",
"updated": "2011-01-01T01:02:03Z"
}

### 1.2.6.31 List images

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/images/detail</td>
<td>Lists images.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the list images request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. List images: JSON response**

```
{
  "images": [
```
1.2.6.32 Create server with configuration drive

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers</td>
<td>Creates a server with the configuration drive extended attribute.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

**Caution**

- When an instance created using a Linux image is started or restarted, the sshd settings are initialized, and it may no longer be possible to log in to the instance. Refer to "Reboot server" on page 26 for details on the required tasks.
- When the password of the user who registered (imported) the image is changed, the image can no longer be used. In such a case, convey either of the following to the operator and request a change of image.
  - The location of the image following the change of image UUID and user authentication information
  - The image UUID that was registered (imported) following the change of image UUID and user authentication information

**Request**

This table shows the URI parameters for the create server with configuration drive request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the create server with configuration drive request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>security_group</td>
<td>string (Optional)</td>
<td>One or more security_group objects. Specify the name of the security group in the name attribute. If you omit this attribute, the server is created in the default security group. Specify a security group for which TCP communication is permitted for the following IP address and port number. IP address: 169.254.169.254 Port number: 80 If TCP communication is not permitted, the host name (computer name) and administrator password may not be set when creating an instance.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>user_data</td>
<td>string</td>
<td>Configuration information or scripts to use upon launch. Must be Base64 encoded. The main formats that are supported are as follows:</td>
</tr>
<tr>
<td></td>
<td>(Optional)</td>
<td>• Linux:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Shell script (begins with #!)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Windows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PowerShell (begins with #ps1_sysnative or #ps1_x86)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Windows batch (begins with rem cmd)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If Linux, cloud-config files can also be specified, but since verification has not been completed, we recommend using Shell scripts.</td>
</tr>
<tr>
<td>availability_zone</td>
<td>string</td>
<td>The availability zone in which to launch the server.</td>
</tr>
<tr>
<td></td>
<td>(Optional)</td>
<td></td>
</tr>
<tr>
<td>server</td>
<td>ServerForCreate</td>
<td>server.</td>
</tr>
<tr>
<td>imageRef</td>
<td>string</td>
<td>The image reference for the desired image for your server instance. Specify as an ID or full URL.</td>
</tr>
<tr>
<td>flavorRef</td>
<td>string</td>
<td>The flavor reference for the desired flavor for your server instance. Specify as an ID or full URL.</td>
</tr>
<tr>
<td>key_name</td>
<td>string</td>
<td>Assigns the public key of the named keypair to the server.</td>
</tr>
<tr>
<td>(Optional)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>networks</td>
<td>string</td>
<td>A networks object. By default, the server instance is provisioned with all isolated networks for the tenant.</td>
</tr>
<tr>
<td>(Optional)</td>
<td></td>
<td>Optionally, you can create one or more NICs on the server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To provision the server instance with a NIC for a network, specify the UUID of the network in the uuid attribute in a networks object.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To provision the server instance with a NIC for an already existing port, specify the port-id in the port attribute in a networks object.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You can specify multiple NICs on the server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ensure that the network is specified.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To set the following information for a virtual server, a virtual router must be connected to the network that the virtual server will be connected to.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Host name (computer name)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Administrator password</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Authentication key (key pair)</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-----------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>uuid</td>
<td>string (Optional)</td>
<td>To provision the server instance with a NIC for a network, specify the UUID of the network in the uuid attribute in a networks object. Required if you omit the port attribute. If the network includes multiple subnets, IP addresses will be allocated from any subnet. To allocate an IP address of a specific subnet, create a port in advance, and specify the uuid of that port in port.</td>
</tr>
<tr>
<td>port</td>
<td>string (Optional)</td>
<td>To provision the server instance with a NIC for an already existing port, specify the port-id in the port attribute in a networks object. Required if you omit the uuid attribute.</td>
</tr>
<tr>
<td>fixed_ip</td>
<td>string (Optional)</td>
<td>A fixed IPv4 address for the NIC. Valid with a neutron or nova-networks network.</td>
</tr>
<tr>
<td>name</td>
<td>string</td>
<td>The server name. This information is also used as the computer name/host name. If 64 characters or more are specified:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Windows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The computer name is the default name set by Windows.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Linux:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The host name will be &quot;host-fixedIpAddressOfEth0&quot;.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The string set for the computer name/host name is changed as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Spaces ( ) and underscores (_) are replaced with hyphens (-)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Uppercase letters are replaced with lowercase letters</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Symbols other than periods (.) and hyphens (-) are removed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Periods (.) are removed from the beginning and end of the string, and strings consisting of hyphens (-) are removed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• In Windows, if the string contains a period (.) anywhere other than at the beginning or end, then the characters preceding the period (.) will be used for the name.</td>
</tr>
<tr>
<td>metadata</td>
<td>string (Optional)</td>
<td>Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each. Windows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To specify a password of an instance, specify the &quot;admin_pass&quot; key.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The specified password will be set for users specified in cloudbase-init.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>--------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Check with the image provider regarding users who are specified for cloudbase-init. Example setting: &quot;metadata&quot;: {&quot;admin_pass&quot;: &quot;passwordSetForInstance&quot;} To use the automatic failover feature, specify &quot;fcx.autofailover&quot;: &quot;true&quot;.</td>
</tr>
<tr>
<td>block_device_mapping_v2</td>
<td>string</td>
<td>Enables booting the server from a volume when additional parameters are given.</td>
</tr>
<tr>
<td>device_name</td>
<td>string</td>
<td>Describes a path to the device for the volume you want to use to boot the server. Specify this item in /dev/vd/deviceName format. /dev/vd is fixed, and for deviceName, specify characters that are valid as a device name. When creating an instance that is allocated multiple volumes, for the boot volume, specify the character with highest priority among the device names of all volumes. The order of priority is a &gt; b &gt; c &gt; ...</td>
</tr>
<tr>
<td>source_type</td>
<td>string</td>
<td>Describes the volume source type for the volume. Choices are &quot;snapshot&quot;, &quot;volume&quot;, or &quot;image&quot;.</td>
</tr>
<tr>
<td>destination_type</td>
<td>string</td>
<td>Specifies the connection destination (&quot;volume&quot;).</td>
</tr>
<tr>
<td>delete_on_termination</td>
<td>bool</td>
<td>(Optional) Specifies whether volumes created during the instance creation will be deleted when the instance is deleted. When &quot;True&quot; is specified, volumes during instance creation will also be deleted when the instance is deleted. When &quot;False&quot; is specified, the volumes created during instance creation will not be deleted when the instance is deleted. If not specified, &quot;False&quot; (do not delete) will be used. The volume where snapshots are collected will not be deleted even if &quot;True&quot; is specified.</td>
</tr>
<tr>
<td>boot_index</td>
<td>string</td>
<td>Specifies the device start order. Specify sequential values, starting from 0. For the boot disk, specify &quot;0&quot;.</td>
</tr>
<tr>
<td>config_drive</td>
<td>string</td>
<td>(Optional) Only &quot;false&quot; can be specified.</td>
</tr>
<tr>
<td>uuid</td>
<td>string</td>
<td>Specifies the UUID of the resource specified for source_type.</td>
</tr>
<tr>
<td>volume_size</td>
<td>string</td>
<td>Specify the volume size in GB. This item must be specified when &quot;image&quot; is specified for source_type. Specify a value equal to or higher than the min_disk parameter of the image to be used. If the min_disk parameter of the image to be used has not been specified or</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>----------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>is &quot;0&quot;</strong>, check the minimum size with the image provider and specify the value accordingly. If &quot;volume&quot; was specified for source_type, this item will be ignored even if a value is specified. If &quot;snapshot&quot; was specified for source_type, and this item is omitted, the volume size of the snapshot collection source will be used.</td>
</tr>
<tr>
<td>OS-DCF:diskConfig</td>
<td>string (Optional)</td>
<td>A valid value is AUTO or MANUAL.</td>
</tr>
</tbody>
</table>

**Example 3.394. Create server with configuration drive: JSON request**

```
{
  "server": {
    "name": "new-server-test",
    "imageRef": "b5660a6e-4b46-4be3-9707-6b47221b454f",
    "flavorRef": "2",
    "key_name": "keypair1",
    "metadata": {
      "My Server Name": "Apache1"
    }
  }
}
```

**Response**

**Example. Create server with configuration drive: JSON response**

```
{
  "server": {
    "adminPass": "am5LKVsBVQ4s",
    "id": "58da039c-dc81-4d8f-8688-a2f819e2f750",
    "links": [
      {
        "href": "http://openstack.example.com/v2/openstack/servers/58da039c-dc81-4d8f-8688-a2f819e2f750",
        "rel": "self"
      },
      {
        "href": "http://openstack.example.com/openstack/servers/58da039c-dc81-4d8f-8688-a2f819e2f750",
        "rel": "bookmark"
      }
    ]
  }
}
```
1.2.6.33 Get server information with configuration drive

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/{server_id}</td>
<td>Shows information for a specified server including the configuration drive extended attribute.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the get server information with configuration drive request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>Uuid</td>
<td>The UUID for the server of interest to you.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. Get server information with configuration drive: JSON response

```json
{
    "server": {
        "accessIPv4": ",
        "accessIPv6": ",
        "addresses": {
            "private": [
                {"addr": "192.168.0.3", "version": 4}
            ]
        },
        "config_drive": ",
        "created": "2013-02-04T13:17:50Z",
        "flavor": {
            "id": "1",
            "links": [
                {"href": "http://openstack.example.com/openstack/flavors/1", "rel": "bookmark"}
            ]
        },
        "hostId": "8725fb615b191d8249a40f3e90d1efde88d914412e4ed2719176afd",
        "id": "dd3b0715-a3fc-43d8-bbd2-2720beb226fb",
        "image": {
            "id": "70a599e0-31e7-49b7-b260-868f441e862b",
            "links": [
                {"href": "http://openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b", "rel": "bookmark"}
            ]
        }
    }
}
```
1.2.6.34 Get server details with configuration drive

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/detail</td>
<td>Lists details for all servers including the configuration drive extended attribute.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the get server details with configuration drive request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. Get server details with configuration drive: JSON response

```json
{
  "servers": [
    {
      "accessIPv4": "",
      "accessIPv6": "",
      "addresses": {
        "private": [
          { "addr": "192.168.0.3", "version": 4 }
        ]
      }
    }
  ]
}
```
1.2.6.35 Create server with OS-EXT-IPS-MAC:mac_addr extended attribute

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers</td>
<td>Creates a server with the OS-EXT-IPS-MAC:mac_addr extended attribute.</td>
</tr>
</tbody>
</table>

Normal response codes: 202
Caution

• When an instance created using a Linux image is started or restarted, the sshd settings are initialized, and it may no longer be possible to log in to the instance. Refer to "Reboot server on page 26" for details on the required tasks.

• When the password of the user who registered (imported) the image is changed, the image can no longer be used. In such a case, convey either of the following to the operator and request a change of image.
  • The location of the image following the change of image UUID and user authentication information
  • The image UUID that was registered (imported) following the change of image UUID and user authentication information

Request

This table shows the URI parameters for the create server with os-ext-ips-mac:mac_addr extended attribute request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the create server with os-ext-ips-mac:mac_addr extended attribute request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>security_group</td>
<td>string</td>
<td>One or more security_group objects. Specify the name of the security group in the name attribute. If you omit this attribute, the server is created in the default security group. Specify a security group for which TCP communication is permitted for the following IP address and port number. IP address: 169.254.169.254 Port number: 80 If TCP communication is not permitted, the host name (computer name) and administrator password may not be set when creating an instance.</td>
</tr>
</tbody>
</table>
| user_data        | string        | Configuration information or scripts to use upon launch. Must be Base64 encoded. The main formats that are supported are as follows:  
  • Linux:  
    • Shell script (begins with #!)  
  • Windows:  
    • PowerShell (begins with #ps1_sysnative or #ps1_x86)  
    • Windows batch (begins with rem cmd) |
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>availability_zone</td>
<td>string (Optional)</td>
<td>The availability zone in which to launch the server.</td>
</tr>
<tr>
<td>server</td>
<td>ServerForCreate</td>
<td>server.</td>
</tr>
<tr>
<td>imageRef</td>
<td>string</td>
<td>The image reference for the desired image for your server instance. Specify as an ID or full URL.</td>
</tr>
<tr>
<td>flavorRef</td>
<td>string</td>
<td>The flavor reference for the desired flavor for your server instance. Specify as an ID or full URL.</td>
</tr>
<tr>
<td>key_name</td>
<td>string (Optional)</td>
<td>Assigns the public key of the named keypair to the server.</td>
</tr>
</tbody>
</table>
| networks     | string (Optional) | A networks object. By default, the server instance is provisioned with all isolated networks for the tenant.  
Optionally, you can create one or more NICs on the server.  
To provision the server instance with a NIC for a network, specify the UUID of the network in the uuid attribute in a networks object.  
To provision the server instance with a NIC for an already existing port, specify the port-id in the port attribute in a networks object.  
You can specify multiple NICs on the server.  
Ensure that the network is specified.  
To set the following information for a virtual server, a virtual router must be connected to the network that the virtual server will be connected to.  
- Host name (computer name)  
- Administrator password  
- Authentication key (key pair) |
| uuid         | string (Optional) | To provision the server instance with a NIC for a network, specify the UUID of the network in the uuid attribute in a networks object. Required if you omit the port attribute.  
If the network includes multiple subnets, IP addresses will be allocated from any subnet. To allocate an IP address of a specific subnet, create a port in advance, and specify the uuid of that port in port. |
<p>| port         | string (Optional) | To provision the server instance with a NIC for an already existing port, specify the port-id in the |</p>
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>port attribute</td>
<td></td>
<td>A port attribute in a networks object. Required if you omit the uuid attribute.</td>
</tr>
<tr>
<td>fixed_ip</td>
<td>string (Optional)</td>
<td>A fixed IPv4 address for the NIC. Valid with a neutron or nova-networks network.</td>
</tr>
<tr>
<td>name</td>
<td>string</td>
<td>The server name.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This information is also used as the computer name/host name.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If 64 characters or more are specified:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Windows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The computer name is the default name set by Windows.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Linux:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The host name will be &quot;host-fixedIpAddressOfEth0&quot;.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The string set for the computer name/host name is changed as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Spaces ( ) and underscores (_) are replaced with hyphens (-)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Uppercase letters are replaced with lowercase letters</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Symbols other than periods (.) and hyphens (-) are removed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Periods (.) are removed from the beginning and end of the string, and strings consisting of hyphens (-) are removed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• In Windows, if the string contains a period (.) anywhere other than at the beginning or end, then the characters preceding the period (.) will be used for the name.</td>
</tr>
<tr>
<td>metadata</td>
<td>string (Optional)</td>
<td>Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To specify a password of an instance, specify the &quot;admin_pass&quot; key.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The specified password will be set for users specified in cloudbase-init.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Check with the image provider regarding users who are specified for cloudbase-init.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example setting: &quot;metadata&quot;: {&quot;admin_pass&quot;: &quot;passwordSetForInstance&quot;}</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To use the automatic failover feature, specify &quot;fcx.autofailover&quot;: &quot;true&quot;.</td>
</tr>
<tr>
<td>block_device_mapping_v2</td>
<td>string</td>
<td>Enables booting the server from a volume when additional parameters are given.</td>
</tr>
<tr>
<td>device_name</td>
<td>string</td>
<td>Describes a path to the device for the volume you want to use to boot the server.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>deviceName</td>
<td>string</td>
<td>Specify this item in `/dev/vd/deviceName format. /dev/vd is fixed, and for deviceName, specify characters that are valid as a device name.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>When creating an instance that is allocated multiple volumes, for the boot volume, specify the character with highest priority among the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>device names of all volumes. The order of priority is a &gt; b &gt; c &gt; ...</td>
</tr>
<tr>
<td>source_type</td>
<td>string</td>
<td>Describes the volume source type for the volume. Choices are &quot;snapshot&quot;, &quot;volume&quot;, or &quot;image&quot;.</td>
</tr>
<tr>
<td>destination_type</td>
<td>string</td>
<td>Specifies the connection destination (&quot;volume&quot;).</td>
</tr>
<tr>
<td>delete_on_termination</td>
<td>bool</td>
<td>Specifies whether volumes created during the instance creation will be deleted when the instance is deleted.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>When &quot;True&quot; is specified, volumes during instance creation will also be deleted when the instance is deleted.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>When &quot;False&quot; is specified, the volumes created during instance creation will not be deleted when the instance is deleted.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If not specified, &quot;False&quot; (do not delete) will be used.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The volume where snapshots are collected will not be deleted even if &quot;True&quot; is specified.</td>
</tr>
<tr>
<td>boot_index</td>
<td>string</td>
<td>Specifies the device start order. Specify sequential values, starting from 0. For the boot disk, specify &quot;0&quot;.</td>
</tr>
<tr>
<td>config_drive</td>
<td>string</td>
<td>Only &quot;false&quot; can be specified.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional)</td>
</tr>
<tr>
<td>uuid</td>
<td>uuid</td>
<td>Specifies the UUID of the resource specified for source_type.</td>
</tr>
<tr>
<td>volume_size</td>
<td>String</td>
<td>Specify the volume size in GB.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This item must be specified when &quot;image&quot; is specified for source_type. Specify a value equal to or higher than the min_disk parameter of</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the image to be used. If the min_disk parameter of the image to be used has not been specified or is &quot;0&quot;, check the minimum size with the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>image provider and specify the value accordingly.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If &quot;volume&quot; was specified for source_type, this item will be ignored even if a value is specified.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If &quot;snapshot&quot; was specified for source_type, and this item is omitted, the volume size of the snapshot collection source will be used.</td>
</tr>
<tr>
<td>OS-EXT-IPS-MAC:mac_addr</td>
<td>string</td>
<td>Associated MAC address.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional)</td>
</tr>
</tbody>
</table>
Example. Create server with OS-EXT-IPS-MAC:mac_addr extended attribute: JSON request

```json
{
  "server": {
    "name": "new-server-test",
    "imageRef": "b5660a6e-4b46-4be3-9707-6b47221b454f",
    "flavorRef": "2",
    "key_name": "keypair1",
    "metadata": {
      "My Server Name": "Apache1" 
    }
  }
}
```

Response

Example. Create server with OS-EXT-IPS-MAC:mac_addr extended attribute: JSON response

```json
{
  "server": {
    "adminPass": "zD7wDKTXiHsp",
    "id": "b44e5008-42f7-4048-b4c8-f40a29da88ba",
    "links": [
      {
        "href": "http://openstack.example.com/v2/openstack/servers/b44e5008-42f7-4048-b4c8-f40a29da88ba",
        "rel": "self"
      },
      {
        "href": "http://openstack.example.com/openstack/servers/b44e5008-42f7-4048-b4c8-f40a29da88ba",
        "rel": "bookmark"
      }
    ]
  }
}
```

1.2.6.36 Show server information

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/{server_id}</td>
<td>Shows information for a specified server. Includes the OSEXT-IPS-MAC:mac_addr extended attribute.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the show server information request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
</tbody>
</table>
This operation does not require a request body.

Response

Example. Show server information: JSON response

```json
{
  "server": {
    "accessIPv4": "",
    "accessIPv6": "",
    "addresses": {
      "private": [
        {
          "addr": "192.168.0.3",
          "version": 4,
          "OS-EXT-IPS-MAC:mac_addr": "00:0c:29:e1:42:90"
        }
      ],
      "created": "2013-02-07T18:46:28Z",
      "flavor": {
        "id": "1",
        "links": [
          {
            "href": "http://openstack.example.com/openstack/flavors/1",
            "rel": "bookmark"
          }
        ],
        "hostId": "4e2003eddbfd1280c2618d04090bced6773203b8da8347af0b2723d",
        "id": "dc7281f9-ee47-40b9-9950-9f73e7961caa",
        "image": {
          "id": "70a599e0-31e7-49b7-b260-868f441e862b",
          "links": [
            {
              "href": "http://openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
              "rel": "bookmark"
            }
          ],
          "links": [
            {
              "href": "http://openstack.example.com/v2/openstack/servers/dc7281f9-ee47-40b9-9950-9f73e7961caa",
              "rel": "self"
            },
            {
              "href": "http://openstack.example.com/openstack/servers/dc7281f9-ee47-40b9-9950-9f73e7961caa",
              "rel": "bookmark"
            }
          ],
          "metadata": {
            "My Server Name": "Apache1"
          },
          "name": "new-server-test",
          "progress": 0,
          "status": "ACTIVE",
          "tenant_id": "openstack",
          "updated": "2013-02-07T18:46:29Z",
          "user_id": "fake"
        }
      }
    }
  }
}
```
1.2.6.37 Get server details

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/detail</td>
<td>Lists details for all servers. Includes the OS-EXT-IPSMAC: mac_addr extended attribute.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the get server details request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. Get server details: JSON response

```json
{
  "servers": [
    {
      "accessIPv4": "",
      "accessIPv6": "",
      "addresses": {
        "private": [
          {
            "addr": "192.168.0.3",
            "version": 4,
            "OS-EXT-IPS-MAC:mac_addr": "00:0c:29:e1:42:90"
          }
        ],
      },
      "created": "2013-02-07T18:40:59Z",
      "flavor": {
        "id": "1",
        "links": [
          {
            "href": "http://openstack.example.com/openstack/flavors/1",
            "rel": "bookmark"
          }
        ],
      },
      "hostId": "fe866a4962fe3bdc2db9c8f7dcdb9555aca73387e72b5cb9c45bd3",
      "id": "76908712-653a-4d16-807e-d89d41435d24",
      "image": {
        "id": "70a599e0-31e7-49b7-b260-868f441e862b",
        "links": [
          {
            "href": "http://openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
            "rel": "bookmark"
          }
        ],
      },
    }
  ]
}
```
1.2.6.38 Create server

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers</td>
<td>Creates a server with a block device mapping.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

**Caution**

- When an instance created using a Linux image is started or restarted, the sshd settings are initialized, and it may no longer be possible to log in to the instance. Refer to "Reboot server" on page 26 for details on the required tasks.
  - When the password of the user who registered (imported) the image is changed, the image can no longer be used. In such a case, convey either of the following to the operator and request a change of image.
    - The location of the image following the change of image UUID and user authentication information
    - The image UUID that was registered (imported) following the change of image UUID and user authentication information

**Request**

This table shows the URI parameters for the create server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the create server request:
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>security_group</td>
<td>string (Optional)</td>
<td>One or more security_group objects. Specify the name of the security group in the name attribute. If you omit this attribute, the server is created in the default security group. Specify a security group for which TCP communication is permitted for the following IP address and port number. IP address: 169.254.169.254 Port number: 80 If TCP communication is not permitted, the host name (computer name) and administrator password may not be set when creating an instance.</td>
</tr>
<tr>
<td>user_data</td>
<td>string (Optional)</td>
<td>Configuration information or scripts to use upon launch. Must be Base64 encoded. The main formats that are supported are as follows: • Linux: • Shell script (begins with #!) • Windows: • PowerShell (begins with #ps1_sysnative or #ps1_x86) • Windows batch (begins with rem cmd) If Linux, cloud-config files can also be specified, but since verification has not been completed, we recommend using Shell scripts.</td>
</tr>
<tr>
<td>availability_zone</td>
<td>string (Optional)</td>
<td>The availability zone in which to launch the server.</td>
</tr>
<tr>
<td>server</td>
<td>ServerForCreate</td>
<td>server.</td>
</tr>
<tr>
<td>imageRef</td>
<td>string</td>
<td>The image reference for the desired image for your server instance. Specify as an ID or full URL.</td>
</tr>
<tr>
<td>flavorRef</td>
<td>string</td>
<td>The flavor reference for the desired flavor for your server instance. Specify as an ID or full URL.</td>
</tr>
<tr>
<td>key_name</td>
<td>string (Optional)</td>
<td>Assigns the public key of the named keypair to the server.</td>
</tr>
<tr>
<td>networks</td>
<td>string (Optional)</td>
<td>A networks object. By default, the server instance is provisioned with all isolated networks for the tenant. Optionally, you can create one or more NICs on the server. To provision the server instance with a NIC for a network, specify the UUID of the network in the uuid attribute in a networks object.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-----------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To provision the server instance with a NIC for an already existing port, specify the port-id in the port attribute in a networks object.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You can specify multiple NICs on the server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ensure that the network is specified.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To set the following information for a virtual server, a virtual router must be connected to the network that the virtual server will be connected to.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Host name (computer name)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Administrator password</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Authentication key (key pair)</td>
</tr>
<tr>
<td>uuid</td>
<td>string</td>
<td>To provision the server instance with a NIC for a network, specify the UUID of the network in the uuid attribute in a networks object. Required if you omit the port attribute.</td>
</tr>
<tr>
<td></td>
<td>(Optional)</td>
<td>If the network includes multiple subnets, IP addresses will be allocated from any subnet. To allocate an IP address of a specific subnet, create a port in advance, and specify the uuid of that port in port.</td>
</tr>
<tr>
<td>port</td>
<td>string</td>
<td>To provision the server instance with a NIC for an already existing port, specify the port-id in the port attribute in a networks object. Required if you omit the uuid attribute.</td>
</tr>
<tr>
<td></td>
<td>(Optional)</td>
<td></td>
</tr>
<tr>
<td>fixed_ip</td>
<td>string</td>
<td>A fixed IPv4 address for the NIC. Valid with a neutron or nova-networks network.</td>
</tr>
<tr>
<td></td>
<td>(Optional)</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>string</td>
<td>The server name.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This information is also used as the computer name/host name.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If 64 characters or more are specified:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Windows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The computer name is the default name set by Windows.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Linux:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The host name will be &quot;host-fixedIpAddressOfEth0&quot;.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The string set for the computer name/host name is changed as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Spaces ( ) and underscores (_) are replaced with hyphens (-)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Uppercase letters are replaced with lowercase letters</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Symbols other than periods (.) and hyphens (-) are removed</td>
</tr>
</tbody>
</table>

- 88 -
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>• Periods (.) are removed from the beginning and end of the string, and strings consisting of hyphens (-) are removed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• In Windows, if the string contains a period (.) anywhere other than at the beginning or end, then the characters preceding the period (.) will be used for the name.</td>
</tr>
<tr>
<td>metadata</td>
<td>string (Optional)</td>
<td>Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To specify a password of an instance, specify the &quot;admin_pass&quot; key.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The specified password will be set for users specified in cloudbase-init.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Check with the image provider regarding users who are specified for cloudbase-init.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example setting: &quot;metadata&quot;: {&quot;admin_pass&quot;: &quot;passwordSetForInstance&quot;}</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To use the automatic failover feature, specify &quot;fcx.autofailover&quot;: &quot;true&quot;.</td>
</tr>
<tr>
<td>block_device_mapping_v2</td>
<td>string</td>
<td>Enables booting the server from a volume when additional parameters are given.</td>
</tr>
<tr>
<td>device_name</td>
<td>string</td>
<td>Describes a path to the device for the volume you want to use to boot the server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Specify this item in /dev/vd/deviceName format. /dev/vd is fixed, and for deviceName, specify characters that are valid as a device name.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>When creating an instance that is allocated multiple volumes, for the boot volume, specify the character with highest priority among the device names of all volumes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The order of priority is a &gt; b &gt; c &gt; ...</td>
</tr>
<tr>
<td>source_type</td>
<td>String</td>
<td>Describes the volume source type for the volume. Choices are &quot;snapshot&quot;, &quot;volume&quot;, or &quot;image&quot;.</td>
</tr>
<tr>
<td>destination_type</td>
<td>String</td>
<td>Specifies the connection destination (<em>volume</em>).</td>
</tr>
<tr>
<td>delete_on_termination</td>
<td>bool (Optional)</td>
<td>Specifies whether volumes created during the instance creation will be deleted when the instance is deleted.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>When &quot;True&quot; is specified, volumes during instance creation will also be deleted when the instance is deleted.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>When &quot;False&quot; is specified, the volumes created during instance creation will not be deleted when the instance is deleted.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If not specified, &quot;False&quot; (do not delete) will be used.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>boot_index</td>
<td>String</td>
<td>Specifies the device start order. Specify sequential values, starting from 0. For the boot disk, specify &quot;0&quot;.</td>
</tr>
<tr>
<td>config_drive</td>
<td>String (Optional)</td>
<td>Only &quot;false&quot; can be specified.</td>
</tr>
<tr>
<td>uuid</td>
<td>uuid</td>
<td>Specifies the UUID of the resource specified for source_type.</td>
</tr>
<tr>
<td>volume_size</td>
<td>String</td>
<td>Specify the volume size in GB. This item must be specified when &quot;image&quot; is specified for source_type. Specify a value equal to or higher than the min_disk parameter of the image to be used. If the min_disk parameter of the image to be used has not been specified or is &quot;0&quot;, check the minimum size with the image provider and specify the value accordingly. If &quot;volume&quot; was specified for source_type, this item will be ignored even if a value is specified. If &quot;snapshot&quot; was specified for source_type, and this item is omitted, the volume size of the snapshot collection source will be used.</td>
</tr>
</tbody>
</table>

**Example. Create server: JSON request**

```json
{
    "server": {
        "name": "new-server-test",
        "imageRef": "b5660a6e-4b46-4be3-9707-6b47221b454f",
        "flavorRef": "2",
        "key_name": "keypair1",
        "metadata": {
            "My Server Name": "Apache1"
        },
        "block_device_mapping_v2": [
            {
                "device_name": "/dev/vda",
                "source_type": "image",
                "destination_type": "volume",
                "volume_size": "20",
                "boot_index": "0",
                "uuid": "6cbf9710-87e3-4a36-8116-9b3396882621",
                "delete_on_termination": "True"
            },
            {
                "device_name": "/dev/vdb",
                "source_type": "volume",
                "destination_type": "volume",
                "boot_index": "1",
                "uuid": "0a273d8d-c5e1-4886-bd93-1d1779283fa3",
                "delete_on_termination": "True"
            },
            {
                "device_name": "/dev/vdc",
                "source_type": "snapshot",
```
"destination_type": "volume",
"volume_size": "30",
"boot_index": "2",
"uuid": "492eac4d-6c12-4828-b0ec-75d3bff0bd4b",
"delete_on_termination": "True"
}
]
]}

Response

Example. Create server: JSON response

```json
{
  "server": {
    "adminPass": "N4x7wFX6iN8D",
    "id": "babd1af0-4fc6-4529-b32f-aad69811ccf5",
    "links": [
      {
        "href": "http://openstack.example.com/v2/openstack/servers/babd1af0-4fc6-4529-b32f-aad69811ccf5",
        "rel": "self"
      },
      {
        "href": "http://openstack.example.com/openstack/servers/babd1af0-4fc6-4529-b32f-aad69811ccf5",
        "rel": "bookmark"
      }
    ]
  }
}
```

1.2.6.39 List servers

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers{?changes-since,image,flavor,name,status,host}</td>
<td>Lists IDs, names, and links for all servers.</td>
</tr>
</tbody>
</table>

Normal response codes: 200, 203

Request

This table shows the URI parameters for the list servers request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This table shows the query parameters for the list servers request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>changes-since</td>
<td>DateTime (Optional)</td>
<td>A time/date stamp for when the image last changed status.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>image</td>
<td>AnyURI</td>
<td>Name of the image in URL format.</td>
</tr>
<tr>
<td></td>
<td>(Optional)</td>
<td></td>
</tr>
<tr>
<td>flavor</td>
<td>AnyURI</td>
<td>Name of the flavor in URL format.</td>
</tr>
<tr>
<td></td>
<td>(Optional)</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the server as a string.</td>
</tr>
<tr>
<td></td>
<td>(Optional)</td>
<td></td>
</tr>
<tr>
<td>status</td>
<td>Server Status</td>
<td>Value of the status of the server so that you can filter on &quot;ACTIVE&quot; for example.</td>
</tr>
<tr>
<td></td>
<td>(Optional)</td>
<td></td>
</tr>
<tr>
<td>host</td>
<td>String</td>
<td>Name of the host as a string.</td>
</tr>
<tr>
<td></td>
<td>(Optional)</td>
<td></td>
</tr>
</tbody>
</table>

Response

Example. List servers: JSON response

```json
{
  "servers": [ 
    {
      "id": "616fb98f-46ca-475e-917e-2563e5a8cd19",
      "links": [ 
        {
          "href": "http://openstack.example.com/v2/openstack/servers/616fb98f-46ca-475e-917e-2563e5a8cd19",
          "rel": "self"
        },
        {
          "href": "http://openstack.example.com/openstack/servers/616fb98f-46ca-475e-917e-2563e5a8cd19",
          "rel": "bookmark"
        }
      ],
      "name": "new-server-test"
    }
  ]
}
```

1.2.6.40 Attach volume

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers/{server_id}/os-volume_attachments</td>
<td>Attaches a volume to the specified server.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the attach volume request:
### Name | Type | Description
--- | --- | ---
{tenant_id} | String | The ID for the tenant or account in a multi-tenancy cloud.
{server_id} | UUID | The UUID for the server of interest to you.
{volumeId} | String | ID of the volume to attach.
{device} | String | Name of the device such as, /dev/vdb. Use "auto" for auto-assign (if supported).
{volumeAttachment} | String | A dictionary representation of a volume attachment.

This table shows the body parameters for the attach volume request:

### Name | Type | Description
--- | --- | ---
volumeld | String | ID of the volume to attach.
device | String | Name of the device such as, /dev/vdb. Use "null" for auto-assign (if supported).

**Example. Attach volume: JSON request**

```json
{
    "volumeAttachment": {
        "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
        "device": null
    }
}
```

**Response**

**Example. Attach volume: JSON response**

```json
{
    "volumeAttachment": {
        "device": "/dev/vdd",
        "id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
        "serverId": "0c92f3f6-c253-4c9b-bd43-e880a8d2eb0a",
        "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f803"
    }
}
```

### 1.2.6.41 List volume attachments

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/{server_id}/os-volume_attachments</td>
<td>Lists the volume attachments for a specified server.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the list volume attachments request:
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. List volume attachments: JSON response**

```json
{
    "volumeAttachments": [
        {
            "device": "/dev/sdd",
            "id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
            "serverId": "4d8c3732-a248-40ed-bebc-539a6fffd25c0",
            "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f803"
        },
        {
            "device": "/dev/sdc",
            "id": "a26887c6-c47b-4654-abb5-dfadf7d3f804",
            "serverId": "4d8c3732-a248-40ed-bebc-539a6fffd25c0",
            "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f804"
        }
    ]
}
```

**1.2.6.42 Show volume attachment details**

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/{server_id}/os-volume_attachments/{attachment_id}</td>
<td>Shows details for the specified volume attachment.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the show volume attachment details request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
<tr>
<td>{attachment_id}</td>
<td>String</td>
<td>Volume attachment ID.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. Show volume attachment details: JSON response**

```json
{
}
```
1.2.6.43 Delete volume attachment

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DELETE</td>
<td>/v2/{tenant_id}/servers/{server_id}/os-volume_attachments/{attachment_id}</td>
<td>Deletes the specified volume attachment from the specified server.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

⚠️ The system volume cannot be removed from the instance.

Request

This table shows the URI parameters for the delete volume attachment request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
<tr>
<td>{attachment_id}</td>
<td>String</td>
<td>Volume attachment ID.</td>
</tr>
</tbody>
</table>

This operation does not accept a request body and does not return a response body.

1.2.6.44 Start server

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers/{server_id}/action</td>
<td>Starts a stopped server and changes its status to ACTIVE.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

Request

This table shows the URI parameters for the start server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
</tbody>
</table>
Example. Start server: JSON request

```
{
  "os-start": null
}
```

Response

This operation does not return a response body.

1.2.6.45 Stop server

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers/{server_id}/action</td>
<td>Stops the specified running server and changes its status to STOPPED.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

Caution

When an instance created using a Linux image is started, the sshd settings are initialized, and it may no longer be possible to log in to the instance. Refer to "Reboot server on page 26" and perform the task before stopping the instance, if necessary.

Request

This table shows the URI parameters for the stop server request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
</tbody>
</table>

Example. Stop server: JSON request

```
{
  "os-stop": null
}
```

Response

This operation does not return a response body.

1.2.6.46 List flavors with access type

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/flavors</td>
<td>Lists flavors and includes the access type, which is public or private.</td>
</tr>
</tbody>
</table>

Normal response codes: 200
**Request**

This table shows the URI parameters for the list flavors with access type request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. List flavors with access type: JSON response**

```json
{
  "flavors": [
    {
      "disk": 1,
      "id": "1",
      "links": [
        {
          "href": "http://openstack.example.com/v2/openstack/flavors/1",
          "rel": "self"
        },
        {
          "href": "http://openstack.example.com/openstack/flavors/1",
          "rel": "bookmark"
        }
      ],
      "name": "m1.tiny",
      "os-flavor-access:is_public": true,
      "ram": 512,
      "vcpus": 1
    },
    {
      "disk": 20,
      "id": "2",
      "links": [
        {
          "href": "http://openstack.example.com/v2/openstack/flavors/2",
          "rel": "self"
        },
        {
          "href": "http://openstack.example.com/openstack/flavors/2",
          "rel": "bookmark"
        }
      ],
      "name": "m1.small",
      "os-flavor-access:is_public": true,
      "ram": 2048,
      "vcpus": 1
    },
    {
      "disk": 40,
      "id": "3",
      "links": [
        {
          "href": "http://openstack.example.com/v2/openstack/flavors/3",
          "rel": "self"
        },
        {
          "href": "http://openstack.example.com/openstack/flavors/3",
          "rel": "bookmark"
        }
      ]
    }
  ]
}```
1.2.6.47 Show flavor access type

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/flavors/{flavor_id}</td>
<td>Gets the flavor access type, which is public or private.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the show flavor access type request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{flavor_id}</td>
<td>UUID</td>
<td>The ID of the flavor of interest to you.</td>
</tr>
</tbody>
</table>
This operation does not require a request body.

Response

Example. Show flavor access type: JSON response

```json
{
  "flavor": {
    "disk": 1,
    "id": "1",
    "links": [
      {
        "href": "http://openstack.example.com/v2/openstack/flavors/1",
        "rel": "self"
      },
      {
        "href": "http://openstack.example.com/openstack/flavors/1",
        "rel": "bookmark"
      }
    ],
    "name": "m1.tiny",
    "os-flavor-access:is_public": true,
    "ram": 512,
    "vcpus": 1
  }
}
```

1.2.6.48 Create interface

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers/{server_id}/os-interface</td>
<td>Creates and uses a port interface to attach the port to a server instance.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the create interface request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the create interface request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>interfaceAttachment</td>
<td>String</td>
<td>Specify the interfaceAttachment action in the request body.</td>
</tr>
<tr>
<td>port_id</td>
<td>UUID</td>
<td>The ID of the port for which you want to create an interface.</td>
</tr>
</tbody>
</table>
Example. Create interface: JSON request

```json
{
  "interfaceAttachment": {
    "port_id": "ce531f90-199f-48c0-816c-13e38010b442"
  }
}
```

Response

Example. Create interface: JSON response

```json
{
  "interfaceAttachment": {
    "fixed_ips": [{
      "ip_address": "192.168.1.1",
      "subnet_id": "f8a6e8f8-c2ec-497c-9f23-da9616de54ef"
    }],
    "mac_addr": "fa:16:3e:4c:2c:30",
    "net_id": "3cb9bc59-5699-4588-a4b1-b87f9678bc6",
    "port_id": "ce531f90-199f-48c0-816c-13e38010b442",
    "port_state": "ACTIVE"
  }
}
```

1.2.6.49 List interfaces

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/{server_id}/os-interface</td>
<td>Lists port interfaces.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the list interfaces request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. List interfaces: JSON response

```json
{
  "interfaceAttachments": [
    {
      "port_state": "ACTIVE"
    }
  ]
}
"fixed_ips": [
    {
        "subnet_id": "f8a6e8f8-c2ec-497c-9f23-da9616de54ef",
        "ip_address": "192.168.1.3"
    }
],
"net_id": "3cb9bc59-5699-4588-a4b1-b87f96708bc6",
"port_id": "ce531f90-199f-48c0-816c-13e38010b442",
"mac_addr": "fa:16:3e:4c:2c:30"
]
}

1.2.6.50 Show attached interface information

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/{server_id}/os-interface/{attachment_id}</td>
<td>Shows information about a specified port interface.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the show attached interface information request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
<tr>
<td>{attachment_id}</td>
<td>UUID</td>
<td>The interface ID.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. Show attached interface information: JSON response

```json
{
    "interfaceAttachment": {
        "port_state": "ACTIVE",
        "fixed_ips": [
            {
                "subnet_id": "f8a6e8f8-c2ec-497c-9f23-da9616de54ef",
                "ip_address": "192.168.1.3"
            }
        ],
        "net_id": "3cb9bc59-5699-4588-a4b1-b87f96708bc6",
        "port_id": "ce531f90-199f-48c0-816c-13e38010b442",
        "mac_addr": "fa:16:3e:4c:2c:30"
    }
}
```
1.2.6.51 Detach interface

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DELETE</td>
<td>/v2/{tenant_id}/servers/{server_id}/os-interface/{attachment_id}</td>
<td>Detaches a specified port interface.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

Request

This table shows the URI parameters for the detach interface request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
<tr>
<td>{attachment_id}</td>
<td>UUID</td>
<td>The interface ID.</td>
</tr>
</tbody>
</table>

This operation does not accept a request body and does not return a response body.

1.2.6.52 List server groups

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/os-server-groups</td>
<td>Lists server groups.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the list server group request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. List server groups: JSON response

```json
{
    "server_groups": [
    {
        "id": "616fb98f-46ca-475e-917e-2563e5a8cd19",
        "name": "test",
        "policies": [
            "anti-affinity"
        ],
        "members": [],
        "metadata": {}
    }
    ]
}
```
### 1.2.6.53 Create server group

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/os-server-groups</td>
<td>Create server group.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the create server group request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>The server group name. A non-empty string with no leading or trailing spaces. Maximum length is 255 characters.</td>
</tr>
<tr>
<td>policies</td>
<td>Dict</td>
<td>A list of one or more policy names to associate with the server group. The list must contain at least one policy name. Each policy name must be a non-empty string with no leading or trailing spaces. Maximum length is 255 characters.</td>
</tr>
<tr>
<td>availability_zone</td>
<td>String</td>
<td>Specifies the availability zone where the server group will be created. If omitted, the availability zone will be determined automatically from the UUID of the domain that the request execution user belongs to.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the create server group request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>The server group name. A non-empty string with no leading or trailing spaces. Maximum length is 255 characters.</td>
</tr>
<tr>
<td>policies</td>
<td>Dict</td>
<td>A list of one or more policy names to associate with the server group. The list must contain at least one policy name. Each policy name must be a non-empty string with no leading or trailing spaces. Maximum length is 255 characters.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>---------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>availability_zone</td>
<td>String (Optional)</td>
<td>Specifies the availability zone where the server group will be created. If omitted, the availability zone will be determined automatically from the UUID of the domain that the request execution user belongs to.</td>
</tr>
</tbody>
</table>

**Example. Create server group: JSON request**

```json
{
  "server_group": {
    "name": "test",
    "policies": ["anti-affinity"
  ]
}
}
```

**Response**

**Example. Create server group: JSON response**

```json
{
  "server_group":{
    "members":[],
    "metadata":{},
    "id":"03a54e57-4fcc-40bc-b532-6426a238ee70",
    "policies":[
      "anti-affinity"
    ],
    "name":"test"
  }
}
```

### 1.2.6.54 Show server group details

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/os-server-groups/ {ServerGroup_id}</td>
<td>Shows details for a specified server group.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the show server group request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>{ServerGroup_id}</td>
<td>String</td>
<td>The server group id.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. Show server group: JSON response**

```json
{
   "server_group": {
      "id": "616fb98f-46ca-475e-917e-2563e5a8cd19",
      "name": "test",
      "policies": ["anti-affinity"],
      "members": [],
      "metadata": {},
      "availability_zone": "jp-east-1a"
   }
}
```

### 1.2.6.55 Delete server group

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DELETE</td>
<td>/v2/{tenant_id}/os-server-groups/ {ServerGroup_id}</td>
<td>Deletes a specified server group.</td>
</tr>
</tbody>
</table>

Normal response codes: 204

**Request**

This table shows the URI parameters for the list server group request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{ServerGroup_id}</td>
<td>String</td>
<td>The server group id.</td>
</tr>
</tbody>
</table>

This operation does not accept a request body and does not return a response body.

### 1.2.6.56 Shelve server

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers/{server_id}/action</td>
<td>Shelves a running server.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

**Caution**

The system volume and expansion volume attached to the instance that is to be shelved remain assigned to the applicable instance, without being released.
Request

This table shows the URI parameters for the show shelve server:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
<tr>
<td>shelve</td>
<td>String</td>
<td>Specify the shelve action in the request body.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Example. Shelve server : JSON request

```
{
  "shelve": null
}
```

Response

This operation does not return a response body.

1.2.6.57 Restore shelved server

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers/{server_id}/action</td>
<td>Restores a shelved server.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

⚠️ Caution: This also restores the system volume and expansion volume that were attached to the instance when it was shelved.

Request

This table shows the URI parameters for the show shelve server:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
<tr>
<td>unshelve</td>
<td>String</td>
<td>Specify the unshelve action in the request body.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Example. Restore shelved server : JSON request

```
{
  "unshelve": null
}
```
### 1.2.6.58 Update server metadata items

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers/{server_id}/metadata</td>
<td>Updates metadata items by key for a specified server.</td>
</tr>
</tbody>
</table>

Replaces items that match the specified keys and does not modify items not specified in the request.

If this operation exceeds the metadata items quota, the API throws an overLimit (413) fault.

Normal response codes: 202

### Request

This table shows the URI parameters for the update server metadata items request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>UUID</td>
<td>The tenant ID in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the update server metadata items request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>metadata</td>
<td>hash</td>
<td>A set of key/value pairs. These pair replace any existing key/value pairs in the resources metadata with matching keys. Any key/value pairs in the parameter with keys that do not occur in the existing resource metadata are added to the resources metadata.</td>
</tr>
</tbody>
</table>

**Example. Update server metadata items: JSON request**

```json
{
    "metadata": {
        "name": "test_server"
    }
}
```

**Response**

**Example. Update server metadata items: JSON response**

```json
{
    "metadata": {
        "name": "test_server"
    }
}
```
1.3 Dedicated instances

1.3.1 Restrictions

This feature is not supported as at the time of issue of this document.

1.3.2 How to check whether an instance is dedicated

1.3.2.1 How to check whether an instance is dedicated

Information indicating whether an instance is dedicated is displayed in the API below.
The information that indicates a dedicated instance is as follows.

<table>
<thead>
<tr>
<th>key</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS-SCH-HNT:scheduler_hints</td>
<td>True</td>
</tr>
</tbody>
</table>

1.3.2.2 Show server information

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/servers/{server_id}</td>
<td>Shows information for a specified server.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the show server information request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. Show server information: JSON response

```json
{
    "server": {
        "OS-DCF:diskConfig": "AUTO",
        "accessIPv4": "",
        "accessIPv6": "",
        "addresses": {
            "private": [
                {
                    "addr": "192.168.0.3",
                }
            ]
        }
    }
}
```
1.3.2.3 List servers

Method | URI | Description |
--- | --- | --- |
GET | /v2/{tenant_id}/servers/detail | Lists servers. |

Normal response codes: 200, 203

Request

This table shows the URI parameters for the list servers request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.
Response

Example. List servers: JSON response

```json
{
  "servers": [
    {
      "OS-DCF:diskConfig": "AUTO",
      "accessIPv4": "",
      "accessIPv6": "",
      "addresses": {
        "private": [
          {
            "addr": "192.168.0.3",
            "version": 4
          }
        ]
      },
      "created": "2012-12-02T02:11:55Z",
      "flavor": {
        "id": "1",
        "links": [
          {
            "href": "http://openstack.example.com/openstack/flavors/1",
            "rel": "bookmark"
          }
        ]
      },
      "hostId": "99428f32351a5d89d0f7727c6ec68c1777c545a0972aaac645508dc",
      "id": "05372e62-05b9-4ee2-9343-9a1fdf2a5fda",
      "image": {
        "id": "70a599e0-31e7-49b7-b260-868f441e862b",
        "links": [
          {
            "href": "http://openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
            "rel": "bookmark"
          }
        ]
      },
      "links": [
        {
          "href": "http://openstack.example.com/v2/openstack/servers/05372e62-05b9-4ee2-9343-9a1fdf2a5fda",
          "rel": "self"
        },
        {
          "href": "http://openstack.example.com/openstack/servers/05372e62-05b9-4ee2-9343-9a1fdf2a5fda",
          "rel": "bookmark"
        }
      ],
      "metadata": {
        "My Server Name": "Apache1"
      },
      "name": "new-server-test",
      "OS-SCH-HNT:scheduler_hints": {
        "fcx.dedicated": "True"
      },
      "progress": 0,
      "status": "ACTIVE",
      "tenant_id": "openstack",
      "updated": "2012-12-02T02:11:56Z",
      "user_id": "fake"
    }
  ]
}
```
1.4 Provisioning script

1.4.1 Provisioning script

Use the APIs listed in *API list* on page 7
1.5 Auto scale

1.5.1 Generate URLs when using APIs

For URLs to be used by the APIs (items 1 and 2), use URLs of the "orchestration" type from the Service catalog retrieved from the identity service.

The endpoint URL is returned in the following format by the identity service.

```
https://hostName/v1/{tenant_id}
```

Create the URL in one of the following formats:

- If you remember the tenant_id:
  Join the path name of each API in the host section of the end point URL.
- If you do not remember the tenant_id:
  The endpoint URL and the path name of each API with "v2/{tenant_id}" removed from the beginning.

For URLs to be used by the APIs (others than items 1 and 2), use URLs of the "autoscale" type from the Service catalog retrieved from the identity service.

The endpoint URL is returned in the following format by the identity service.

```
https://hostName/autoscale_schedulers
```

Join the path name of each API in the host section of the end point URL, and create the URL.

1.5.2 API list

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>POST /v1/{tenant_id}/stacks/{stack_name}/stack_id/resources/{resource_name}/signal</td>
<td>Sends signal to the specified resource</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Send signal</td>
</tr>
<tr>
<td>2</td>
<td>GET /v1/{tenant_id}/stacks/{stack_name}/stack_id/resources/{resource_name}</td>
<td>Retrieves detailed information about the specified resource</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Retrieve details of stack resources</td>
</tr>
<tr>
<td>3</td>
<td>POST /autoscale_schedulers</td>
<td>Registers a schedule</td>
</tr>
<tr>
<td>4</td>
<td>DELETE /autoscale_schedulers/{name}</td>
<td>Deletes a schedule.</td>
</tr>
<tr>
<td>5</td>
<td>GET /autoscale_schedulers</td>
<td>Retrieves a schedule list</td>
</tr>
</tbody>
</table>
For processing other than the above, use the following APIs described in 「API Reference Manual (Application Platform Service)」 — 「Template/Development environment」 — 「Orchestration」

- Create stack: POST /v1/{tenant_id}/stacks (Create a stack)
- Find stack: GET /v1/{tenant_id}/stacks/{stack_name} (Retrieve the URL of the specified stack)
- Update stack: PUT /v1/{tenant_id}/stacks/{stack_name}/{stack_id} (Update the specified stack)
- Delete stack: DELETE /v1/{tenant_id}/stacks/{stack_name} (Delete the specified stack)

1.5.3 Request header

<table>
<thead>
<tr>
<th>Header</th>
<th>Specified value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accept</td>
<td>application/json</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>X-Auth-Token</td>
<td>Authentication token</td>
</tr>
</tbody>
</table>

1.5.4 API error codes

<table>
<thead>
<tr>
<th>Error code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>500, other codes possible</td>
<td>Server Error, cloudServersFault</td>
</tr>
<tr>
<td>400</td>
<td>BadRequest</td>
</tr>
<tr>
<td>401</td>
<td>unauthorized</td>
</tr>
<tr>
<td></td>
<td>This code is returned by the schedule API when its execution is not approved under the operator privileges (role).</td>
</tr>
<tr>
<td>403</td>
<td>Forbidden, resizeNotAllowed</td>
</tr>
<tr>
<td>404</td>
<td>itemNotFound</td>
</tr>
<tr>
<td>405</td>
<td>BadMethod</td>
</tr>
<tr>
<td>409</td>
<td>conflictingRequest</td>
</tr>
<tr>
<td>413</td>
<td>OverLimit</td>
</tr>
<tr>
<td>415</td>
<td>badMediaType</td>
</tr>
<tr>
<td>501</td>
<td>notImplemented</td>
</tr>
<tr>
<td>503</td>
<td>serviceUnavailable</td>
</tr>
</tbody>
</table>
1.5.5 API details

1.5.5.1 Send signal

Method: POST
URI: /v1/{tenant_id}/stacks/{stack_name}/
     {stack_id}/resources/{resource_name}/signal
Description: Sends a signal to the specified resource.
Normal response code: 200

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tenant_id</td>
<td>String</td>
<td>Tenant ID</td>
</tr>
<tr>
<td>stack_name</td>
<td>String</td>
<td>Stack name</td>
</tr>
<tr>
<td>stack_id</td>
<td>String</td>
<td>Stack ID</td>
</tr>
<tr>
<td>resource_name</td>
<td>String</td>
<td>Stack resource name</td>
</tr>
</tbody>
</table>

Example request
No request body

Example response
No response body

1.5.5.2 Retrieve details of stack resources

Method: GET
URI: /v1/{tenant_id}/stacks/{stack_name}/
     {stack_id}/resources/{resource_name}
Description: Retrieves details of the specified resource.
Normal response code: 200

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tenant_id</td>
<td>String</td>
<td>Tenant ID</td>
</tr>
<tr>
<td>stack_name</td>
<td>String</td>
<td>Stack name</td>
</tr>
<tr>
<td>stack_id</td>
<td>String</td>
<td>Stack ID</td>
</tr>
<tr>
<td>resource_name</td>
<td>String</td>
<td>Stack resource name</td>
</tr>
</tbody>
</table>

Example request
No request body
Example response

```json
{
   "resource": {
      "resource_name": "web_server_group",
      "description": "",
      "links": [
         {"href": "http://heatServiceIpAddress:portNumber/v1/a0b06965de4a43569795bb0feadb0856/stacks/test_stack/0cc8b90c-0163-42fe-a261-4b005fd89b2a/resources/web_server_group", "rel": "self"},
         {"href": "http://heatServiceIpAddress:portNumber/v1/a0b06965de4a43569795bb0feadb0856/stacks/test_stack/0cc8b90c-0163-42fe-a261-4b005fd89b2a","rel": "stack"}
      ],
      "logical_resource_id": "web_server_group",
      "resource_status": "CREATE_COMPLETE",
      "updated_time": "2014-09-16T15:41:42Z",
      "required_by": ["web_server_scaleout_policy", "web_server_scalein_policy"],
      "resource_status_reason": "state changed",
      "physical_resource_id": "f4ff3753-a52c-4b2f-bd95-600344922a1a",
      "resource_type": "OS::Heat::AutoScalingGroup"
   }
}
```

1.5.5.3 Register a schedule

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/autoscale_schedulers</td>
<td>Registers a schedule</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Caution**
- Registration of schedules must be performed by the same user who created the stack for executing schedules.
- In addition to the above condition, registration of schedules when using custom roles must be performed by a user for whom the orch_scheduler_create and orch_create roles were set.
- While the schedule feature is being used, do not delete the roles specified for the user when creating stacks and registering schedules.

**Request**

This table shows the URI parameters for the request body.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Specify a name for identifying the schedule. A unique name is required for all tenants in a region. Specify a name using up to 64 alphanumeric characters (a-z, A-Z, 0-9). The name is case-sensitive.</td>
</tr>
<tr>
<td>method</td>
<td>String</td>
<td>Specify POST.</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>Specify the signal URL.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Specify a name for identifying the schedule. A unique name is required for all tenants in a region.</td>
</tr>
<tr>
<td>time</td>
<td>String</td>
<td>Specify a datetime for executing the REST API. The format must comply with the crom command.</td>
</tr>
<tr>
<td>project_id</td>
<td>String</td>
<td>Specify the ID of the project for which schedule processing will be executed.</td>
</tr>
</tbody>
</table>

**Example. Create schedule: JSON request**

```json
{
  "name": "sample2",
  "time": "*/5 * * * *",
  "url": "http://orchestration.jp-east-1.cloud.global.fujitsu.com/v1/2801e5de00d142a388b17b65256f7ad4/stacks/s1_step2/0ffc2720-a7ba-4e0f-8ca6-82142f14f6eb/resources/web_server_scaleup_policy/signal",
  "method": "POST",
  "project_id": "2801e5de00d142a388b17b65256f7ad4"
}
```

**Response**

**Example. Create server group: JSON response**

```json
{
  "name": "sample2",
  "method": "POST",
  "url": "http://orchestration.jp-east-1.cloud.global.fujitsu.com/v1/2801e5de00d142a388b17b65256f7ad4/stacks/s1_step2/0ffc2720-a7ba-4e0f-8ca6-82142f14f6eb/resources/web_server_scaleup_policy/signal",
  "time": "*/5 * * * *",
  "project_id": "2801e5de00d142a388b17b65256f7ad4"
}
```

### 1.5.5.4 Delete a schedule

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DELETE</td>
<td>/autoscaleSchedulers/{name}</td>
<td>Deletes a schedule.</td>
</tr>
</tbody>
</table>

Normal response codes: 204

**Request**

This table shows the URI parameters for the request body.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Specify a name for identifying the schedule. A unique name is required for all tenants in a region.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Specify a name using up to 64 alphanumeric characters (a-z, A-Z, 0-9). The name is case-sensitive.</td>
</tr>
</tbody>
</table>

There are no parameters to specify in the request body.

**Response**
There is no request body.

### 1.5.5.5 List schedules

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/autoscale_schedulers</td>
<td>Retrieves a schedule list</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**
There are no parameters to specify in the request URI and body.

**Response**

**Example. List schedules: JSON response**

```json
{
    "schedulers": [
        {
            "name": "sample2",
            "time": "/5 * * * *",
            "method": "POST",
            "url": "http://192.168.3.104:8004/v1/2801e5de00d142a388b17b65256f7ad4/
                      stacks/s1_step2/0fbb2720-a7ba-4e0f-8ca6-82142f14f6eb/resources/
                      web_server_scaleup_policy(signal",
            "project_id": "2801e5de00d142a388b17b65256f7ad4"
        }
    ]
}
```
1.6 Virtual server imports

1.6.1 Import a virtual server

Each API can be run using the Administrator or System Owner role.

1.6.2 API list

<table>
<thead>
<tr>
<th>Item</th>
<th>API name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>POST /v1/imageimport</td>
<td>Registers an image</td>
</tr>
<tr>
<td>2</td>
<td>GET /v1/imageimport/{import_id}/status</td>
<td>Retrieves the status of image registration</td>
</tr>
<tr>
<td>3</td>
<td>GET /v1/imageimport/?start, limit</td>
<td>Retrieves the status of the image registration list</td>
</tr>
</tbody>
</table>

Note: {import_id} is the import ID returned by the virtual server import request API.

1.6.3 API details

1.6.3.1 Register image

URL
/v1/imageimport

Method
POST

Request header
Refer to "Table 1: Request headers on page 119" for details on the request headers of this API.

Table 1: Request headers

<table>
<thead>
<tr>
<th>Request header</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-Auth-Token</td>
<td>String</td>
<td>Tokens that were retrieved using the Identity service</td>
</tr>
</tbody>
</table>
## Request body

Refer to "Table 2: Request parameters" on page 120 for details on the request parameters of this API. Each parameter is specified in JSON format in the request body.

### Table 2: Request parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Specifies the image name. (*1)</td>
</tr>
<tr>
<td>disk_format (optional)</td>
<td>String</td>
<td>Specifies the storage disk format for services (*1). If &quot;true&quot; is specified for the &quot;conversion&quot; parameter, you must specify &quot;raw&quot; for this parameter.</td>
</tr>
<tr>
<td>container_format (optional)</td>
<td>String</td>
<td>Specifies the storage container format for services. Only &quot;bare&quot; can be specified. (*1)</td>
</tr>
</tbody>
</table>
| location           | String| Part of the URL of object storage where VM image files from the transfer source are deployed. Specify using the following format: 
| checksum (optional) | String| The SHA1 checksum value of the VM image file from the transfer source. If omitted, checksum verification will not be performed when importing a virtual server. |
| id                 | String| Specifies the UUID to be allocated to an image for registration. Specify using the following format: 
| min_ram (optional) | int   | Specifies the minimum RAM (MB) required for using images. If omitted, 0 will be used. (*1) |
| min_disk (optional) | int   | Specifies the minimum disk size (GB) required for using images. If omitted, 0 will be used. (*1) |
| property-* (optional) |       | Specifies the property to be set for an image. If omitted, a property will not be set for an image. For the **"** in the parameter name, specify the property key name. (*1) |
| ovf_location (optional) | String| Part of the URL of object storage where ovf files are deployed for VM image files from the transfer source. Specify using the following format: 
| conversion         | boolean| Specifies whether images are converted (*2). Only "true" can be specified. |
| os_type            | String| OS type in VM image at transfer source. Specify one of the following. |

(*1) Required parameter

(*2) If "true" is specified for the "conversion" parameter, "raw" must be specified for the "disk_format" parameter. 

Only "bare" can be specified for the "container_format" parameter. 

If "true" is specified for the "conversion" parameter, you must specify "raw" for the "disk_format" parameter.
### Table 3: List of response parameters when the state is normal

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>import_id</td>
<td>String</td>
<td>ID of the import process. Used when retrieving the conversion status.</td>
</tr>
</tbody>
</table>

*1: Parameter registered for a service.

*2: Images in vmdk format are converted into a format that can be processed by K5 (raw format).

### Response body

Refer to "Table 3: List of response parameters when the state is normal" on page 121" and "Table 4: List of response parameters when the state is error" on page 121" for details on the response parameters of this API. Each parameter is specified in JSON format in the response body. Refer to "Table 5: Detailed error information list" on page 122" for details on error information returned when an error occurs.

**Table 3: List of response parameters when the state is normal**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>import_id</td>
<td>String</td>
<td>ID of the import process. Used when retrieving the conversion status.</td>
</tr>
</tbody>
</table>

**Table 4: List of response parameters when the state is error**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>error_code</td>
<td>String</td>
<td>Detailed error code</td>
</tr>
<tr>
<td>message</td>
<td>String</td>
<td>Detailed error message</td>
</tr>
</tbody>
</table>
Table 5: Detailed error information list

<table>
<thead>
<tr>
<th>Status code</th>
<th>Detailed error code</th>
<th>Detailed error message</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>400</td>
<td>40001</td>
<td>Required parameter 'name' is not specified.</td>
<td>Modify the request content.</td>
</tr>
<tr>
<td>400</td>
<td>40002</td>
<td>Required parameter 'location' is not specified.</td>
<td>Modify the request content.</td>
</tr>
<tr>
<td>400</td>
<td>40003</td>
<td>Required parameter 'id' is not specified.</td>
<td>Modify the request content.</td>
</tr>
<tr>
<td>400</td>
<td>40004</td>
<td>Required parameter 'conversion' is not specified.</td>
<td>Modify the request content.</td>
</tr>
<tr>
<td>400</td>
<td>40005</td>
<td>Required parameter 'os_type' is not specified.</td>
<td>Modify the request content.</td>
</tr>
<tr>
<td>400</td>
<td>40006</td>
<td>Parameter 'os_type' has invalid value.</td>
<td>Modify the request content.</td>
</tr>
<tr>
<td>400</td>
<td>40007</td>
<td>Required parameter 'user_name' is not specified.</td>
<td>Modify the request content.</td>
</tr>
<tr>
<td>400</td>
<td>40008</td>
<td>Required parameter 'password' is not specified.</td>
<td>Modify the request content.</td>
</tr>
<tr>
<td>400</td>
<td>40009</td>
<td>Required parameter 'domain_name' is not specified.</td>
<td>Modify the request content.</td>
</tr>
<tr>
<td>400</td>
<td>40010</td>
<td>Parameter 'user_name' or 'password' or 'domain_name' has invalid value.</td>
<td>Modify the request content.</td>
</tr>
<tr>
<td>403</td>
<td>40301</td>
<td>You cannot execute this API.</td>
<td>Check with the Administrator about your account privileges.</td>
</tr>
<tr>
<td>500</td>
<td>50001</td>
<td>Internal server error.</td>
<td>Contact support.</td>
</tr>
<tr>
<td>500</td>
<td>50002</td>
<td>Internal server error.</td>
<td>Contact support.</td>
</tr>
<tr>
<td>500</td>
<td>50003</td>
<td>Internal server error.</td>
<td>Contact support.</td>
</tr>
<tr>
<td>503</td>
<td>50301</td>
<td>Too many requests.</td>
<td>Reexecute after completing other import requests.</td>
</tr>
</tbody>
</table>

**Status code**

Refer to *Table 6: Status codes on page 122* for details on the status codes returned by this API.

Table 6: Status codes

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>201</td>
<td>Indicates that the virtual server import process started normally.</td>
</tr>
<tr>
<td>400</td>
<td>Indicates an invalid request.</td>
</tr>
<tr>
<td>401</td>
<td>Indicates an invalid token.</td>
</tr>
<tr>
<td>403</td>
<td>Indicates that the virtual server import execution privileges do not exist.</td>
</tr>
<tr>
<td>500</td>
<td>Indicates that the virtual server import process failed due to an error.</td>
</tr>
</tbody>
</table>
Status code | Description
--- | ---
503 | Indicates that the virtual server import process failed due to the upper limit for concurrent executions of requests being exceeded.

Example response body

An example response body is shown below.

```
{"import_id":"6bf65660-70ee-4a75-b61a-36ac040c289c"}
```

1.6.3.2 Retrieve image registration status

**URL**

/v1/imageimport/{import_id}/status

Note: {import_id} is the import ID returned by the virtual server import request API.

**Method**

GET

**Request header**

Refer to "Table 7: Request headers on page 123" for details on the request headers of this API.

<table>
<thead>
<tr>
<th>Request header</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-Auth-Token</td>
<td>String</td>
<td>Tokens that were retrieved using the Identity service</td>
</tr>
</tbody>
</table>

**Request body**

This API does not accept request parameters. "0" must be specified in the Content-Length header. Any content specified for the request body will be ignored.

**Response body**

Refer to "Table 8: Response parameters on page 123", "Table 9: List of response parameters when import fails on page 124" and "Table 10: Response parameters when an error occurs on page 124" for details on the response parameters of this API. Each parameter is specified in JSON format in the response body. If import_status is "succeeded" or "processing", the content of "Table 8: Response parameters on page 123" is returned. If import_status is "failed", the content of "Table 8: Response parameters on page 123" and "Table 9: List of response parameters when import fails on page 124" is returned. Refer to "Table 11: Detailed error information list on page 125" for details on error information returned when an error occurs.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>import_status</td>
<td>String</td>
<td>Status of the import process. One of the following is returned.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Value</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Keyword</strong></td>
</tr>
<tr>
<td>succeeded:</td>
<td></td>
<td>Indicates that the import process was completed successfully.</td>
</tr>
<tr>
<td>Failed:</td>
<td></td>
<td>Indicates that the import process failed.</td>
</tr>
<tr>
<td>processing:</td>
<td></td>
<td>Indicates that the import process is in progress.</td>
</tr>
<tr>
<td>queued:</td>
<td></td>
<td>Indicates that an import process item is waiting to start.</td>
</tr>
<tr>
<td>progress</td>
<td>int</td>
<td>Progress status of the import process. Value from 0 to 100.</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Image name</td>
</tr>
<tr>
<td>disk_format</td>
<td>String</td>
<td>Format of storage disk for service.</td>
</tr>
<tr>
<td>container_format</td>
<td>String</td>
<td>Format of storage container for service.</td>
</tr>
<tr>
<td>is_public</td>
<td>boolean</td>
<td>Indicates whether images are shared.</td>
</tr>
<tr>
<td>location</td>
<td>String</td>
<td>URL of object storage where VM image files from the transfer source are deployed.</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>Unique ID of image registered for service.</td>
</tr>
<tr>
<td>min_ram</td>
<td>int</td>
<td>The minimum RAM (MB) required for using images.</td>
</tr>
<tr>
<td>min_disk</td>
<td>int</td>
<td>The minimum disk capacity (GB) required for using images.</td>
</tr>
<tr>
<td>properties</td>
<td></td>
<td>Properties set for images.</td>
</tr>
<tr>
<td>ovf_location</td>
<td>String</td>
<td>URL of object storage where ovf files are deployed for VM image files from the transfer source.</td>
</tr>
<tr>
<td>conversion</td>
<td>boolean</td>
<td>Indicates whether images are converted.</td>
</tr>
<tr>
<td>os_type</td>
<td>String</td>
<td>OS type in VM image at transfer source.</td>
</tr>
<tr>
<td>user_name</td>
<td>String</td>
<td>User name</td>
</tr>
<tr>
<td>password</td>
<td>String</td>
<td>Base64-encoded password</td>
</tr>
<tr>
<td>domain_name</td>
<td>String</td>
<td>Domain name</td>
</tr>
</tbody>
</table>

Table 9: List of response parameters when import fails

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>message</td>
<td>String</td>
<td>Detailed error message</td>
</tr>
</tbody>
</table>

Table 10: Response parameters when an error occurs

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>error_code</td>
<td>String</td>
<td>Detailed error code</td>
</tr>
</tbody>
</table>
Table 11: Detailed error information list

<table>
<thead>
<tr>
<th>Status code</th>
<th>Detailed error code</th>
<th>Detailed error message</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>403</td>
<td>40301</td>
<td>You cannot execute this API.</td>
<td>Check with the Administrator about your account privileges.</td>
</tr>
<tr>
<td>404</td>
<td>40401</td>
<td>Invalid import_id.</td>
<td>Modify the request URL.</td>
</tr>
<tr>
<td>500</td>
<td>50001</td>
<td>Internal server error.</td>
<td>Contact support.</td>
</tr>
<tr>
<td>500</td>
<td>50002</td>
<td>Internal server error.</td>
<td>Contact support.</td>
</tr>
</tbody>
</table>

**Status code**

Refer to "Table 12: Status codes on page 125" for details on the status codes returned by this API.

Table 12: Status codes

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Indicates that retrieval of the virtual server import process status was successful.</td>
</tr>
<tr>
<td>401</td>
<td>Indicates an invalid token.</td>
</tr>
<tr>
<td>403</td>
<td>Indicates that the virtual server import execution privileges do not exist.</td>
</tr>
<tr>
<td>404</td>
<td>The import_id specified in the URL is invalid. The status is stored for 7 days after import completion. This status code is returned when the API is run after this period.</td>
</tr>
<tr>
<td>500</td>
<td>Retrieval of the virtual server import status failed.</td>
</tr>
</tbody>
</table>

**Example response body**

An example response body is shown below.

```json
{"import_status":"succeeded","progress":100,"name":"sample_image","disk_format":"raw","container_format":"bare","is_public":false,"location":"/v1/AUTH_790f1c092038413a8dd6771a033b17f5/test_container/test_image.vmdk","id":"14294f46-d9fe-45bb-ba5f-0b93a69c4416","min_ram":0,"min_disk":0,"properties":{},"ovf_location":"/v1/AUTH_790f1c092038413a8dd6771a033b17f5/test_container/test_image.ovf","conversion":true,"os_type":"RHEL","user_name":"testuser","password":"cGFzc3dvcmQ=","domain_name":"testdomain"}
```

### 1.6.3.3 List image registration status

**URL**

/v1/imageimport{?start, limit}
Refer to *Table 13: Request parameters* on page 126* for details on the query parameters of this API.

**Table 13: Request parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>start (optional)</td>
<td>int</td>
<td>Specifies the index of the first item to retrieve. If omitted, the most recent import will be used.</td>
</tr>
<tr>
<td>limit (optional)</td>
<td>int</td>
<td>Specifies the number of images to retrieve. If omitted, all process information corresponding to the specified token will be returned.</td>
</tr>
</tbody>
</table>

**Method**

GET

**Request header**

Refer to *Table 14: Request headers* on page 126* for details on the request headers of this API.

**Table 14: Request headers**

<table>
<thead>
<tr>
<th>Request header</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-Auth-Token</td>
<td>String</td>
<td>Tokens that were retrieved using the Identity service</td>
</tr>
</tbody>
</table>

**Request body**

This API does not accept the request body. "0" must be specified in the Content-Length header. Any content specified for the request body will be ignored.

**Response body**

Refer to *Table 15: Response parameters* on page 126* and *Table 16: Response parameters when an error occurs* on page 127* for details on the response parameters of this API. Refer to *Table 17: Detailed error information list* on page 127* for details on error information returned when an error occurs.

**Table 15: Response parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>imports</td>
<td>Array</td>
<td>The route element of a virtual server import process list. Process information matching the project ID of the specified token is returned as an array element. Up to 100 elements are returned, by request order, starting from the newest.</td>
</tr>
<tr>
<td>import_id</td>
<td>String</td>
<td>ID of the import process.</td>
</tr>
<tr>
<td>import_status</td>
<td>String</td>
<td>Status of the import process. One of the following is returned.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Value</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>Keyword</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Succeeded:</td>
<td>Indicates that the import process was completed successfully.</td>
<td></td>
</tr>
<tr>
<td>Failed:</td>
<td>Indicates that the import process failed.</td>
<td></td>
</tr>
<tr>
<td>Processing:</td>
<td>Indicates that the import process is in progress.</td>
<td></td>
</tr>
<tr>
<td>Queued:</td>
<td>Indicates that the import process is waiting to start.</td>
<td></td>
</tr>
</tbody>
</table>

**status** String URL for retrieving the registration status.

Table 16: Response parameters when an error occurs

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>error_code</td>
<td>String</td>
<td>Detailed error code</td>
</tr>
<tr>
<td>message</td>
<td>String</td>
<td>Detailed error message</td>
</tr>
</tbody>
</table>

Table 17: Detailed error information list

<table>
<thead>
<tr>
<th>Status code</th>
<th>Detailed error code</th>
<th>Detailed error message</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>403</td>
<td>40301</td>
<td>You cannot execute this API.</td>
<td>Check with the Administrator about your account privileges.</td>
</tr>
<tr>
<td>500</td>
<td>50001</td>
<td>Internal server error.</td>
<td>Contact support.</td>
</tr>
<tr>
<td>500</td>
<td>50002</td>
<td>Internal server error.</td>
<td>Contact support.</td>
</tr>
</tbody>
</table>

**Status code**

Refer to "Table 18: Status codes on page 127" for details on the status codes returned by this API.

Table 18: Status codes

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Indicates that retrieval of the virtual server import process status was successful.</td>
</tr>
<tr>
<td>400</td>
<td>Indicates an invalid request.</td>
</tr>
<tr>
<td>401</td>
<td>Indicates an invalid token.</td>
</tr>
<tr>
<td>403</td>
<td>Indicates that the virtual server import execution privileges do not exist.</td>
</tr>
<tr>
<td>500</td>
<td>Indicates that retrieval of the virtual server import process list failed due to an error.</td>
</tr>
</tbody>
</table>
Example response body

An example response body is shown below.

```json
{"imports": [{"import_id": "6bf65660-70ee-4a75-b61a-36ac040c289c", "import_status": "succeeded", "status": "/v1/imageimport/6bf65660-70ee-4a75-b61a-36ac040c289c/status"}]
```
## 1.7 Virtual servers for SAP

### 1.7.1 API list

#### 1.7.1.1 API list

<table>
<thead>
<tr>
<th>Item</th>
<th>API name</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | GET /templates/l_servers
Retrieve a list of virtual server templates | Retrieves a list of virtual server templates |
| 2    | POST /l_servers
Create virtual server | Creates a virtual server |
| 3    | GET /l_servers
Retrieve a list of virtual servers | Retrieves a list of virtual servers |
| 4    | GET /l_servers/resourceId
Retrieve virtual server details | Retrieves detailed information on the specified virtual server. Disk, NIC and snapshot information are included in the retrieved information. |
| 5    | PUT /l_servers/resourceId/start
Start virtual server | Starts the specified virtual server |
| 6    | PUT /l_servers/resourceId/stop
Stop virtual server | Stops the specified virtual server |
| 7    | PUT /l_servers/resourceId/restart
Restart virtual server | Restarts the specified virtual server |
| 8    | DELETE /l_servers/resourceId
Delete virtual server | Deletes the specified virtual server |
| 9    | PUT /l_servers/resourceId
Change virtual server | Changes information about the specified virtual server |
| 10   | PUT /l_servers/resourceId/attach
Attach disk to virtual server | Attaches a disk to the specified virtual server |
| 11   | PUT /l_servers/resourceId/detach
Detach disk from virtual server | Detaches a disk from the specified virtual server |
| 12   | PUT /l_servers/resourceId/attach
Attach NIC to virtual server | Attaches an NIC to the specified virtual server |
| 13   | PUT /l_servers/resourceId/detach
Detach NIC from virtual server | Detaches an NIC from the specified virtual server |
<p>| 14   | POST /server_images | Creates a clone image from a virtual server |</p>
<table>
<thead>
<tr>
<th>Item</th>
<th>API name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>DELETE /server_images/cloneImageName</td>
<td>Deletes a clone image</td>
</tr>
<tr>
<td></td>
<td>Delete clone image</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>GET /server_images</td>
<td>Retrieves a list of clone images</td>
</tr>
<tr>
<td></td>
<td>Retrieve a list of clone images</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>PUT /server_images/cloneImageName/move</td>
<td>Changes the disclosure scope of the specified clone image</td>
</tr>
<tr>
<td></td>
<td>Change disclosure scope of clone image</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>POST /server_images</td>
<td>Creates a snapshot of a virtual server</td>
</tr>
<tr>
<td></td>
<td>Create snapshot</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>PUT /server_images/resourceId/restore</td>
<td>Restores a snapshot of the specified virtual server</td>
</tr>
<tr>
<td></td>
<td>Restore snapshot</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>DELETE /server_images/resourceId</td>
<td>Deletes a snapshot</td>
</tr>
<tr>
<td></td>
<td>Delete snapshot</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>GET /tasks/taskId</td>
<td>Retrieves detailed information about the specified task</td>
</tr>
<tr>
<td></td>
<td>Retrieve task details</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>POST /projects</td>
<td>Enables a project</td>
</tr>
<tr>
<td></td>
<td>Enable project</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>DELETE /projects/projId</td>
<td>Disables the specified project</td>
</tr>
<tr>
<td></td>
<td>Disable project</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>GET /projects/projId</td>
<td>Retrieves detailed information about the specified project</td>
</tr>
<tr>
<td></td>
<td>Retrieve project details</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>POST /networks</td>
<td>Creates a network resource</td>
</tr>
<tr>
<td></td>
<td>Create network resource</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>DELETE /networks/resourceId</td>
<td>Deletes the specified network resource</td>
</tr>
<tr>
<td></td>
<td>Delete network resource</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>GET /networks</td>
<td>Retrieves a list of network resources</td>
</tr>
<tr>
<td></td>
<td>Retrieve a list of network resources</td>
<td></td>
</tr>
</tbody>
</table>

### 1.7.1.2 Permit/prohibit a preset role

The following symbols are used to indicate permission/prohibition of API access in each preset role.

Note: Refer to the role management service specification/API for details on custom roles.

- **A**: Allowed on all projects
- **B**: Allowed only on projects that the user belongs to
- **N**: Not allowed
| Item | API | Operation Description | Method | Content | Export | Update | Delete | Create
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /server</td>
<td>Get a list of virtual servers</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>POST /server</td>
<td>Create virtual server</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>GET /server</td>
<td>Get a list of virtual servers</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>GET /server/VirtualServer</td>
<td>Get virtual server details</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>PUT /server/VirtualServer</td>
<td>Start virtual server</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>PUT /server/VirtualServer</td>
<td>Stop virtual server</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>PUT /server/VirtualServer/Power</td>
<td>Reset virtual server</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>DELETE /server/VirtualServer</td>
<td>Delete virtual server</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>PUT /server/VirtualServer</td>
<td>Change virtual server</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>PUT /server/VirtualServer</td>
<td>Attach disk to virtual server</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>PUT /server/VirtualServer/Resource</td>
<td>Detach disk from virtual server</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>PUT /server/VirtualServer</td>
<td>Attach disk to virtual server</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>PUT /server/VirtualServer/Resource</td>
<td>Detach disk from virtual server</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>POST server</td>
<td>Create disk image</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>DELETE server/image</td>
<td>Delete disk image</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>GET server/image</td>
<td>Get a list of disk images</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>PUT server/image</td>
<td>Create a disk image</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>PUT server/image</td>
<td>Change disk image's name</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>DELETE server/image</td>
<td>Delete disk image</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>DELETE server/image</td>
<td>Delete snapshot</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
</tbody>
</table>
1.7.2 HTTP request

This section describes the elements comprising an HTTP request: HTTP header, HTTP method, and URL.

HTTP header

The following HTTP headers can be specified:

- **Content-Type**
  - Specifies the content type of the HTTP request.
- **Accept**
  - Specifies the content type of the HTTP response.
- **X-Auth-Token**
  - Specifies the token of Keystone.
    - When the system administrator uses the API, the scope specified for the token is ignored.
- **X-VA-Project-Id**
  - Specifies the project ID.
    - This must be specified when the system administrator uses the API. Users cannot specify the project ID when using the API.
- **X-VA-Domain-Id**
  - Specifies the domain ID.
    - This must be specified when the system administrator uses the API. Users cannot specify the project ID when using the API.

HTTP method

Depending on the purpose of the API, the HTTP methods are distinguished as follows:

- **POST**: Create resources
- **GET**: Retrieve resource information
- **PUT**: Operate resources
- **DELETE**: Delete resources
The URL format is as follows:
<protocol>://<host>:<port>/<path> <params>
<protocol>
Transfer protocol (specify "https")
</protocol>

<host>
Host name or IP address of the delivery platform manager
</host>

<port>
Port number (default: 23461)
</port>

<path>
API path. This must be specified in one of the following formats:

/</resourceType>/
Format when creating resources or retrieving information (example: /l_servers/)

/</resourceType>/<resourceId>/
Format when retrieving information on, changing the attributes of, or deleting specific resources (example: /l_servers/100)

/</resourceType>/<resourceId>/<operationName>
Format when performing operations specific to a particular resource (example: /l_servers/100/start)

<params>
API parameters

When the HTTP method is GET or DELETE:
Arguments must be specified for queries. A question mark (?) must be appended to <path>, followed by the arguments. The arguments must be specified in the format <label>=<value> (*1).

When the HTTP method is POST or PUT:
Arguments must be specified in the request body. The arguments must be specified according to the Content-Type value of the HTTP header.

When Content-Type is application/x-www-form-urlencoded:
Arguments must be specified in the format <label>=<value> (*1).

When Content-Type is application/xml:
Arguments must be specified in XML format. If optional elements are omitted, the XML tags must be omitted. It is not possible to specify a format omitting only values.

*1: The format for specifying <label>=<value> is as follows:
<label>=<value>&<label>=<value>, ...

<value> Value

When omitting optional arguments, both <label> and <value> must be omitted. It is not possible to specify (<label>=" etc.) and omit <value> only.
1.7.3 HTTP response

This section describes the elements comprising an HTTP response: status code and body. Responses are broadly divided into the 3 categories below.

When a request is completed normally

The status code "200 OK" is returned. This code is mainly returned by the resource information retrieval API (HTTP request using the GET method). For the body, the requested information is returned using the content type specified in the Accept header.

When a request is received normally and processing starts asynchronously

The status code "202 Accepted" is returned. This code is mainly returned by APIs other than the resource information retrieval API, when asynchronously executing tasks that involve time-consuming processing. For the body, the task information is returned using the content type specified in the Accept header (currently XML only).

```xml
<task>
  <id>taskId</id>
  <status>waiting</status>
  <progress>0</progress>
  <starttime>startTime</starttime>
  <resource id="resourceId"/>
</task>
```

The XML may include information for internal control purposes. Tags included in the body but not defined as API tags will be ignored.

When an error occurs

A status code from 4xx to 5xx is returned when an error occurs during the processing or accepting stages for HTTP requests.

Refer to Status codes in use on page 174 for details.

For the body, the error information is returned using the content type specified in the Accept header (currently XML only).

**When an error occurs in the delivery platform region manager**

```xml
<error>
  <message>msgBody</message>
  <arg>arg1</arg>
  <arg>arg2</arg>
  ...
  <request_id>requestId</request_id>
</error>
```

**When an error occurs in the delivery platform environment**

```xml
<errors>
  <error>
    <code>msgd</code>
    <message>msgBody</message>
    <arg>arg</arg>
    <arg>arg2</arg>
    ...
    <request_id>requestId</request_id>
  </error>
  <AvailabilityZone>availZoneWhereErrorOccurred</AvailabilityZone>
  <cause>
    <product>relatedProdName</product>
    <resource_name>linkedProdResourceName</resource_name>
  </cause>
</errors>
```
<request>infoRequestForRelatedProd</request>
<message>returnInfoOfRelatedProd</message>
</cause>
</error>
...
*1: Multiple error tags will be displayed when errors occur in multiple availability zones.

The message body is in English only, and embedded arguments are expressed as ${number}.
The API user is able to construct the complete message by deploying arguments to the message body.
Example: Message ID=67114, Message body="can not copy file [${0}->${1}]",
Argument="a.txt","b.txt"
=>FJSVrcx:ERROR:67114:can not copy file [a.txt->b.txt]
The cause tag is output when some information (related product name, linked product resource name, information requested for related product) returned from a product linked to this product is included.
The XML may be extended, with the addition of detailed error information, for example. Tags included in the body but not defined as API tags will be ignored.

1.7.4 XML common specification

The common XML specification handled by APIs is as follows.
• When multiple resources of different types become root elements, the <Resources> tag will be used as a root element.

Example: <Resources>
<LServer>...</LServer
<Disk>...</Disk>
</Resources>

• When multiple resources of the same type become root elements, the tag with the plural format the the resource will be used as a root element.

Example: <LServers>
<LServer>...</LServer
<LServer>...</LServer
</LServers>

1.7.5 API details

1.7.5.1 List image registration status

Description
Lists virtual server templates.
Request

Header

Accept: application/xml
X-Auth-Token: Keystone token

Method

GET

Path

/templates/l_servers

Query

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>verbose</td>
<td>To retrieve only some of the information, specify &quot;false&quot;, or to retrieve all of the information, specify &quot;true&quot;. If omitted, &quot;false&quot; is used.</td>
</tr>
</tbody>
</table>

Body

n/a

Response

Status code

200 OK

Body

If verbose=false:

```
<LServerTemplates>
<LServerTemplate name="virtualServerTempl1"/>
<LServerTemplate name="virtualServerTempl2"/>
</LServerTemplates>
```

If verbose=true:

```
<LServerTemplates>
<LServerTemplate name="virtualServerTemplateName1">
  <CPU>
    <CPUPerf>cpuPerformance</CPUPerf>
    <NumOfCPU numOfCpus/NumOfCPU>
  </CPU>
  <Memory>
    <MemorySize>memSize</MemorySize>
  </Memory>
</LServerTemplate>
<LServerTemplate name="virtualServerTempl2">
  ... (information about 2nd template)
</LServerTemplate>
</LServerTemplates>
```
<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>virtualServer</td>
<td>Name of the virtual server</td>
<td>String of up to 32 characters, which may contain halfwidth alphanumeric characters, underscores (_), and hyphens (-), and starts with a halfwidth letter.</td>
</tr>
<tr>
<td>TemplName</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cpuPerformance</td>
<td>CPU performance</td>
<td>Number with up to one decimal place, in GHz</td>
</tr>
<tr>
<td></td>
<td>allocated to the virtual server</td>
<td></td>
</tr>
<tr>
<td>numOfCpus</td>
<td>Number of CPUs</td>
<td>Integer greater than 0</td>
</tr>
<tr>
<td></td>
<td>allocated to the virtual server</td>
<td></td>
</tr>
<tr>
<td>memSize</td>
<td>Amount of memory</td>
<td>Number with up to one decimal place, in GB</td>
</tr>
<tr>
<td></td>
<td>allocated to the virtual server</td>
<td></td>
</tr>
</tbody>
</table>

### 1.7.5.2 Create virtual server

**Description**

Create a virtual server.

**Request**

**Header**

Content-Type: application/xml
Accept: application/xml
X-Auth-Token: Keystone token (*1)
X-VA-Project-Id: Project ID of resource creation destination (*2)
X-VA-Domain-Id: Domain ID of resource creation destination (*2)

*1: The scope specified for the token is ignored when this API is executed by the system administrator.
*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

**Method**

POST

**Path**

/l_servers

**Query**

n/a

**Body**

```xml
<Resources>
  <LServer name="virtualServerName"/>
  <TemplateLink name="virtualServerTemplName"/>
  <ServerImageLink name="imageName"/>
</Resources>
```
<ServerImageProject name="projId"/>
<Disks>
<Disk>
<DiskIndex>diskNum</DiskIndex>
<DiskSize>diskSize</DiskSize>
</Disk>
</Disks>
<NICs>
<NIC>
<NICIndex>nicNum</NICIndex>
<NetworkLinks>
<NetworkLink id="networkId"/>
<IpAddress auto="ipAutoSetup" address="ipAddr"/>
</NetworkLink>
</NetworkLinks>
</NIC>
</NICs>
<OSSetting>
<ComputerName>computerName, hostName</ComputerName>
<AdminPassword>adminPassword</AdminPassword>
<DNSServers>
<DNSServer nic="nicIndex" ip="dnsIpAddress" />
</DNServers>
</OSSetting>
<AvailabilityZone>availZone</AvailabilityZone>
<Dedicated>deployToDedicatedInstance</Dedicated>
</LServer>
</Resources>

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>virtualServerName</td>
<td>Name of the virtual server</td>
<td>Specify a string of up to 64 characters, using alphanumeric characters, hyphens (-), underscores (_), and periods (.). It must start with a halfwidth alphanumeric character.</td>
</tr>
<tr>
<td>virtualServerTemplName</td>
<td>Template name used by the virtual server</td>
<td>Resource name of virtual server templates that have already been created.</td>
</tr>
<tr>
<td>imageName</td>
<td>Clone image name to be distributed to the boot disk of the virtual server</td>
<td>Resource name of cloning images that have already been created.</td>
</tr>
</tbody>
</table>
| projId        | ID of the project that owns the image       | • Public image:
Leave blank.
• Private image:
ID of the project that owns the image.

Because the image name is unique within the project, image names may be duplicated between projects.
Therefore, this information is necessary for identifying which project an image belongs to, when image names are duplicated.

Note

Specify the following items when not using a template or using a customized template
<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>diskNum</td>
<td>Number of the disk to allocate to the virtual server</td>
<td>Specify integers starting from 1. Specify a value from 1 to 55. <strong>Note</strong> The system disk of the specified image is extracted to the system disk of the virtual server. Therefore, even if the disk size is specified for disk number 0 (system disk), that value will not be reflected—it will be ignored.</td>
</tr>
<tr>
<td>diskSize</td>
<td>Size of the disk to be extracted.</td>
<td>Specify a number of up to one decimal place, in GB.</td>
</tr>
<tr>
<td>nicNum</td>
<td>Number that identifies the NIC to allocate to the virtual server</td>
<td>Specify an integer starting from 0 (specify sequential numbers). Specify a value from 0 to 9.</td>
</tr>
<tr>
<td>networkId</td>
<td>ID of the network to which the virtual server connects</td>
<td>ID format</td>
</tr>
<tr>
<td>ipAddr</td>
<td>IP address to allocate to the virtual server (Optional)</td>
<td>• If specifying the IP address directly&lt;br&gt;<code>&lt;IpAddress auto=&quot;false&quot; address=&quot;xxx.xxx.xxx.xxx&quot;/&gt;</code>&lt;br&gt;• If automatically allocating from the address range set for network resources&lt;br&gt;<code>&lt;IpAddress auto=&quot;true&quot;&gt;</code> or&lt;br&gt;omit the IpAddress tag, or&lt;br&gt;omit the auto attribute.</td>
</tr>
<tr>
<td>computerName, hostName</td>
<td>Computer name, host name</td>
<td>Specify from 1 to 15 characters for Windows or from 1 to 63 characters for Linux using alphanumeric characters and hyphens (-). A string consisting of numbers only cannot be specified. If omitted, the name of the virtual server will be used. Underscores (_) and periods (.) are replaced with hyphens (-).</td>
</tr>
<tr>
<td>adminPassword</td>
<td>Administrator password used when the operating system is Windows</td>
<td>Administrator password of the public or private image to be distributed. If a password is not set for the private image to be distributed, the value specified for this parameter will be used. Specify a value of up to 128 characters, using halfwidth alphanumeric characters and symbols. If omitted, an error will occur. Cannot set passwords for administrator users other than the Administrator.</td>
</tr>
<tr>
<td>Item</td>
<td>Description</td>
<td>Details</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------------------------------------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>nicIndex</code></td>
<td>NIC index for which to configure DNS</td>
<td>Index of the NIC for which to configure the IP address of the DNS.</td>
</tr>
<tr>
<td><code>dnsIpAddress</code></td>
<td>IP address of the DNS</td>
<td>Specify the IP address of the DNS to be set for each NIC if the operating system is Windows. If omitted, the IP address of the DNS will be used.</td>
</tr>
<tr>
<td><code>availZone</code></td>
<td>Name of the availability zone where the virtual server will be created</td>
<td>Name of the availability zone. If omitted, the virtual server will be created in an existing availability zone.</td>
</tr>
<tr>
<td><code>deployToDedicatedInstance</code></td>
<td>Deploy to dedicated instance</td>
<td>Specify &quot;true&quot; to deploy the virtual server to a dedicated instance. Specify &quot;false&quot; to not deploy the virtual server to a dedicated instance. If omitted, &quot;false&quot; will be used.</td>
</tr>
</tbody>
</table>

For subnet mask, default gateway, etc., the network resource value specified in NetworkLink of each NIC will be used.

**Response**

**Status code**

202 Accepted

**Body**

Task information

**1.7.5.3 List virtual servers**

**Description**

Lists virtual servers.

**Request**

**Header**

Accept: application/xml

X-Auth-Token: Keystone token (*1)

X-VA-Project-Id: ID of the project that owns the resource (*2)

X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*1: The scope specified for the token is ignored when this API is executed by the system administrator.

*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.
Method
GET

Path
/l_servers

Query
<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>state</td>
<td>Specify &quot;registered&quot; to retrieve information about virtual servers that have already been created. If omitted, information about virtual servers still being created will also be retrieved.</td>
</tr>
</tbody>
</table>

Body
n/a

Response

Body

```xml
<LServers>
  <LServer name="virtualServerName1" id="virtualServerId1">
    <AvailabilityZone>availabilityZone</AvailabilityZone>
    <Dedicated>dedicatedInstance</Dedicated>
  </LServer>
  <LServer name="virtualServerName2" id="virtualServerId2">
    <AvailabilityZone>availabilityZone</AvailabilityZone>
    <Dedicated>deployedToDedicatedInstance</Dedicated>
  </LServer>
</LServers>
```

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>virtualServerId</td>
<td>ID allocated to the virtual server</td>
<td>ID</td>
</tr>
<tr>
<td>availabilityZone</td>
<td>Name of the availability zone where the virtual server exists</td>
<td>Availability zone name</td>
</tr>
</tbody>
</table>
| deployedToDedicatedInstance | Indicates whether a virtual server is a dedicated instance | • If "true": Indicates that the virtual server is a dedicated instance  
                                 |                     | • If "false": Indicates that the virtual server is not a dedicated instance |

1.7.5.4 Retrieve virtual server details

Description
Retrieves detailed information about the specified virtual server.
Request

Header

Accept: application/xml
X-Auth-Token: Keystone token (*1)
X-VA-Project-Id: ID of the project that owns the resource (*2)
X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*1: The scope specified for the token is ignored when this API is executed by the system administrator.
*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

Method

GET

Path

/l_servers/virtualServerResourceId

Query

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
</table>
| internal      | This parameter is used to retrieve the base image ID of a cloning image specified during creation of the virtual server. This parameter is intended for use by system administrators. However, the base image ID can be retrieved even if specified by a regular user (it will not result in error).
|               | • "true"
|               | Retrieves the base image ID
|               | • "false" (optional)
|               | Does not retrieve the base image ID |

Body

n/a

Response

Status code

200 OK

Body

```xml
<Resources>
    <LServer name="virtualServerName" id="virtualServerId"/>
    <TemplateLink name="virtualServerTemplName" />  
    <Disks>
        <Disk name="diskName"/>
        <DiskLink name="diskNameAllocatedToVirtualServer" id="DiskID" />  
        <DiskIndex>diskNum</DiskIndex>
        <DiskSize>diskSize</DiskSize>
```
<DevicePath>devicePath</DevicePath>

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>snapshot Generation</td>
<td>Generation of the snapshot</td>
<td>Numeric characters.</td>
</tr>
<tr>
<td>snapshot Datetime</td>
<td>Snapshot retrieval datetime</td>
<td>Datetime. 2015/06/12-09:44:42</td>
</tr>
<tr>
<td>snapshot ResourceId</td>
<td>Resource ID of the snapshot</td>
<td>ID format</td>
</tr>
<tr>
<td>diskName</td>
<td>Disk name allocated to the virtual server</td>
<td>If a disk (such as a disk connected to RDM of SAP for virtual server or to an unsupported device) that cannot be managed by the delivery platform is connected, an empty value is returned.</td>
</tr>
<tr>
<td>diskId</td>
<td>Resource ID of the disk</td>
<td>Same as above.</td>
</tr>
<tr>
<td>diskSize</td>
<td>Size of the disk</td>
<td>Refer to diskSize of Create virtual server on page 138 for details.</td>
</tr>
<tr>
<td>devicePath</td>
<td>Device path or identifier for the connection to the disk</td>
<td>SCSI:0:0, etc.</td>
</tr>
<tr>
<td>macAddr</td>
<td>MAC address allocated to the NIC of the virtual server</td>
<td>XX:XX:XX:XX:XX:XX</td>
</tr>
<tr>
<td>resource Status</td>
<td>Status of the virtual server</td>
<td>normal, warning, stop, error, fatal, unknown</td>
</tr>
<tr>
<td>Item</td>
<td>Description</td>
<td>Details</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>powerStatus</td>
<td>Power status of the virtual server</td>
<td>on, off, unknown</td>
</tr>
<tr>
<td>baseImageId</td>
<td>Base image ID of a cloning image specified during creation of the virtual server</td>
<td>ID format (up to 32 alphanumeric characters)</td>
</tr>
</tbody>
</table>

Refer to the creation API for details on other elements

### 1.7.5.5 Start virtual server

**Description**

Starts the virtual server (power on).
There is no waiting for the operating system to start.

**Request**

**Header**

- Accept: application/xml
- X-Auth-Token: Keystone token (*1)
- X-VA-Project-Id: ID of the project that owns the resource (*2)
- X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*1: The scope specified for the token is ignored when this API is executed by the system administrator.

*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

**Method**

PUT

**Path**

/l_servers/virtualServerResourceId/start

**Query**

n/a

**Body**

n/a

**Response**

**Status code**

202 Accepted
1.7.5.6 Stop virtual server

Description
Stops the virtual server (power off).

Request

Header
Content-Type: application/x-www-form-urlencoded
Accept: application/xml
X-Auth-Token: Keystone token (*1)
X-VA-Project-Id: ID of the project that owns the resource (*2)
X-VA-Domain-Id: ID of the domain that owns the resource (*2)
*1: The scope specified for the token is ignored when this API is executed by the system administrator.
*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

Method
PUT

Path
/l_servers/virtualServerResourceId/stop

Query
n/a

Body

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>force</td>
<td>Force option</td>
</tr>
<tr>
<td></td>
<td>• &quot;true&quot;</td>
</tr>
<tr>
<td></td>
<td>Performs a forced stop</td>
</tr>
<tr>
<td></td>
<td>• &quot;false&quot; (optional)</td>
</tr>
<tr>
<td></td>
<td>Does not perform a forced stop</td>
</tr>
</tbody>
</table>

Response

Status code
202 Accepted

Body
Task information
1.7.5.7 Restart virtual server

Description
Restarts (reboots) the virtual server.

Request

Header
Content-Type: application/x-www-form-urlencoded
Accept: application/xml
X-Auth-Token: Keystone token (*1)
X-VA-Project-Id: ID of the project that owns the resource (*2)
X-VA-Domain-Id: ID of the domain that owns the resource (*2)
*1: The scope specified for the token is ignored when this API is executed by the system administrator.
*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

Method
PUT

Path
/l_servers/virtualServerResourceId/restart

Query
n/a

Body

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>force</td>
<td>Force option</td>
</tr>
<tr>
<td></td>
<td>• &quot;true&quot;</td>
</tr>
<tr>
<td></td>
<td>Performs a forced restart</td>
</tr>
<tr>
<td></td>
<td>• &quot;false&quot; (optional)</td>
</tr>
<tr>
<td></td>
<td>Does not perform a forced restart</td>
</tr>
</tbody>
</table>

Response

Status code
202 Accepted

Body
Task information
1.7.5.8 Delete virtual server

**Description**
Deletes a virtual server
Stop (power off) the virtual server before deleting it.

**Request**

**Header**
- X-Auth-Token: Keystone token (*1)
- X-VA-Project-Id: ID of the project that owns the resource (*2)
- X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*1: The scope specified for the token is ignored when this API is executed by the system administrator.
*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

**Method**
> DELETE

**Path**
/lservers/virtualServerResourceId

**Query**
n/a

**Body**
n/a

**Response**

**Status code**
202 Accepted

**Body**
Task information

1.7.5.9 Change virtual server

**Description**
Changes the registration information of the specified virtual server.
It is possible to make changes to the name and server specs (number of CPUs, CPU performance, and memory).
To downsize CPU or memory, it is necessary to stop the virtual server first.
Request

Header

Content-Type: application/xml
Accept: application/xml
X-Auth-Token: Keystone token (*1)
X-VA-Project-Id: ID of the project that owns the resource (*2)
X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*1: The scope specified for the token is ignored when this API is executed by the system administrator.
*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

Method

PUT

Path

slash_servers/virtualServerResourceId

Query

n/a

Body

The content will be changed if it is different to the current settings.

<Resources>
  <LServer name="virtualServerName">
    <TemplateLink name="virtualServerTemplName"/>
  </LServer>
</Resources>

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Refer to the creation API</td>
<td>for details</td>
<td></td>
</tr>
</tbody>
</table>

Response

Status code

202 Accepted

Body

Task information
1.7.5.10 Attach disk to virtual server

**Description**
Attaches a disk to the specified virtual server.

**Request**

**Header**
- Content-Type: application/x-www-form-urlencoded
- Accept: application/xml
- X-Auth-Token: Keystone token (*1)
- X-VA-Project-Id: ID of the project that owns the resource (*2)
- X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*1: The scope specified for the token is ignored when this API is executed by the system administrator.

*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

**Method**
PUT

**Path**
/l_servers/virtualServerResourceId/attach

**Query**
n/a

**Body**

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>index</td>
<td>Disk index (optional). Integer starting from 1.</td>
</tr>
<tr>
<td></td>
<td>If omitted, largestDiskNum + 1 will be used.</td>
</tr>
<tr>
<td></td>
<td>Even if the disk numbers are not consecutive, the way of allocating numbers is the same (for example, if the disk numbers are 1,3 and a disk is attached, the disk numbers become 1,3,4)</td>
</tr>
<tr>
<td>disk_size</td>
<td>Disk capacity</td>
</tr>
<tr>
<td></td>
<td>Number with up to one decimal place, in GB.</td>
</tr>
</tbody>
</table>

**Response**

**Status code**
202 Accepted

**Body**
Task information
1.7.5.11 Detach disk from virtual server

**Description**
Detaches the specified disk from the specified virtual server.

*Stop (power off) the virtual server before detaching the disk.*

Note that the disk number of the remaining disks will remain unchanged after detachment (for example: if the disk numbers are 0,1,2 and 1 is detached, the disk numbers become 0,2)

**Request**

**Header**
- Accept: application/xml
- X-Auth-Token: Keystone token (*1)
- X-VA-Project-Id: ID of the project that owns the resource (*2)
- X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*1: The scope specified for the token is ignored when this API is executed by the system administrator.

*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

**Method**
PUT

**Path**
/l_servers/virtualServerResourceId/detach

**Query**
n/a

**Body**

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>disk_rid</td>
<td>Resource ID of the disk to be detached.</td>
</tr>
</tbody>
</table>

**Response**

**Status code**
202 Accepted

**Body**
Task information
1.7.5.12 Attach NIC to virtual server

**Description**
Attaches an NIC to the specified virtual server.
The NIC number is set by assigning an available sequential number starting from 0
(for example: if the NIC numbers are 0,2 and an NIC is attached, NIC numbers become 0,1,2, and
if NIC numbers are 0,1,2 and an NIC is attached, NIC numbers become 0,1,2,3).
Snapshots of the virtual server must be deleted before performing this operation.

**Request**

**Header**
Content-Type: application/x-www-form-urlencoded
Accept: application/xml
X-Auth-Token: Keystone token (*1)
X-VA-Project-Id: ID of the project that owns the resource (*2)
X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*1: The scope specified for the token is ignored when this API is executed by the system administrator.
*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

**Method**
PUT

**Path**
/l_servers/virtualServerResourceId/attach

**Query**
n/a

**Body**

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>Specify &quot;nic&quot;.</td>
</tr>
<tr>
<td>network_rid</td>
<td>Resource ID of the network to which the NIC will be connected</td>
</tr>
<tr>
<td>ip_address</td>
<td>IP address assigned to the NIC (optional) If omitted, an address will be assigned automatically.</td>
</tr>
</tbody>
</table>

**Response**

**Status code**
202 Accepted
1.7.5.13 Detach NIC from virtual server

**Description**
Detaches the specified NIC from the virtual server.

Note that the NIC number of the NICS will remain unchanged after detachment (for example: if the NIC numbers are 0, 1, 2 and NIC number 1 is detached, NIC numbers become 0, 2).

Snapshots of the virtual server must be deleted before performing this operation.

**Request**

**Header**
- Content-Type: application/x-www-form-urlencoded
- Accept: application/xml
- X-Auth-Token: Keystone token (*1)
- X-VA-Project-Id: ID of the project that owns the resource (*2)
- X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*1: The scope specified for the token is ignored when this API is executed by the system administrator.

*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

**Method**
PUT

**Path**
/l_servers/virtualServerResourceIds/detach

**Query**
n/a

**Body**

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>Specify &quot;nic&quot;.</td>
</tr>
<tr>
<td>index</td>
<td>NIC number Integer starting from 0.</td>
</tr>
</tbody>
</table>

**Response**

**Status code**
202 Accepted

**Body**
Task information
1.7.5.14 Create clone image

**Description**

Creates a clone image from a virtual server.

Stop the target virtual server before creating a clone image.

An error will occur if an image with the same name already exists in the project.

**Caution**

The virtual server where the clone image is created must reflect the specific information (computer name, IP address, etc.) specified during creation of the virtual server in the operating system. The specific information will be reflected in the operating system when the virtual server is started.

Therefore, for the virtual server where the clone image is created, specify one that has been started once.

If you create a clone image from a virtual server that has not been started at all, clone image creation will be successful, however, an error will occur when creating a virtual server using that clone image.

**Request**

**Header**

Content-Type: application/x-www-form-urlencoded
Accept: application/xml
X-Auth-Token: Keystone token (*1)
X-VA-Project-Id: Project ID of resource creation destination (*2)
X-VA-Domain-Id: Domain ID of resource creation destination (*2)

*1: The scope specified for the token is ignored when this API is executed by the system administrator.

*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

**Method**

POST

**Path**

/server_images

**Query**

n/a

**Body**

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specifies the clone image name. Specify a string of up to 32 characters using halfwidth alphanumeric characters and underscores (_). It must start with a halfwidth letter.</td>
</tr>
<tr>
<td>Argument name</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>type</td>
<td>Specify &quot;cloning&quot;.</td>
</tr>
<tr>
<td>l_server_rid</td>
<td>Specifies the resource ID of the virtual server where the clone image will be created.</td>
</tr>
<tr>
<td>comment</td>
<td>Specifies comments for the clone image (optional). Specify a string of up to 96 characters using halfwidth or fullwidth characters, except for percent (%), backslash (), double quotation mark (&quot;), and line feed characters.</td>
</tr>
</tbody>
</table>

**Response**

**Status code**

202 Accepted

**Body**

Task information

However, `<resource id="resourceId"/>` is not displayed.

### 1.7.5.15 Delete clone image

**Description**

Deletes a clone image.

**Request**

**Header**

Accept: application/xml
X-Auth-Token: Keystone token (*1)
X-VA-Project-Id: ID of the project that owns the resource (*2)
X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*1: The scope specified for the token is ignored when this API is executed by the system administrator.
*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

**Method**

DELETE

**Path**

`/server_images/cloneImageName`
Query

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>Specify &quot;cloning&quot;.</td>
</tr>
</tbody>
</table>

Body

n/a

Response

Status code

202 Accepted

Body

Returns information on tasks executed by each availability zone. However, `<resource id="resourceId"/>` is not displayed.

```
<tasks>
  <task>
    <id>taskId</id>
    <status>waiting</status>
    <progress>0</progress>
    <starttime>startTime</starttime>
    <AvailabilityZone>availabilityZoneName</AvailabilityZone>
  </task>
  ...
</tasks>
```

### 1.7.5.16 List clone images

Description

Lists images.

Request

Header

- Accept: application/xml
- X-Auth-Token: Keystone token (*1)
- X-VA-Project-Id: ID of the project that owns the resource (*2)
- X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*1: The scope specified for the token is ignored when this API is executed by the system administrator.

*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.
Method
GET

Path
/server_images

Query

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>verbose</td>
<td>To retrieve only some of the information, specify &quot;false&quot;, or to retrieve all of the information, specify &quot;true&quot;. If omitted, &quot;false&quot; is used.</td>
</tr>
<tr>
<td>mode</td>
<td>Specify &quot;cloning&quot;.</td>
</tr>
<tr>
<td>disksize</td>
<td>Specify &quot;true&quot; to retrieve disk information held by an image. Specify &quot;false&quot; to not retrieve disk information held by an image. If omitted, &quot;false&quot; will be used.</td>
</tr>
<tr>
<td>availability_zone</td>
<td>Specifies the name of the availability zone where the list of images will be retrieved from. If omitted, a list of images will be retrieved from all availability zones.</td>
</tr>
</tbody>
</table>

Response

Status code
200 OK

Body
If verbose=false, mode=cloning, and disksize=false:

```xml
<ServerImages>
  <ServerImage name="imageName1">
    <Scope>imageDisclosureScope</Scope>
    <Project>idOfProjWhereImageWasCreated</Project>
    <AvailabilityZone>availZone</AvailabilityZone>
  </ServerImage>
  <ServerImage name="imageName2">
    <Scope>imageDisclosureScope</Scope>
    <Project>idOfProjWhereImageWasCreated</Project>
    <AvailabilityZone>availZone</AvailabilityZone>
  </ServerImage>
</ServerImages>
```

If verbose=true, mode=cloning, and disksize=false:

```xml
<ServerImages>
  <ServerImage name="imageName1">
    <Scope>imageDisclosureScope</Scope>
    <Project>idOfProjWhereImageWasCreated</Project>
    <AvailabilityZone>availZone</AvailabilityZone>
    <Comment>comment</Comment>
    <CreateTime>creationDatetime</CreateTime>
  </ServerImage>
  <ServerImage name="imageName2">
    <Scope>imageDisclosureScope</Scope>
    <Project>idOfProjWhereImageWasCreated</Project>
    <AvailabilityZone>availZone</AvailabilityZone>
  </ServerImage>
</ServerImages>
```
If verbose=true, mode=cloning, and disksize=true:

```xml
<ServerImages>
  <ServerImage name="imageName1">
    <Scope>imageDisclosureScope</Scope>
    <Project>idOfProjWhereImageWasCreated</Project>
    <AvailabilityZone>availZone</AvailabilityZone>
    <Comment>comment</Comment>
    <CreateTime>creationDatetime</CreateTime>
    <Disks>
      <Disk>
        <DiskIndex>diskNum</DiskIndex>
        <DiskSize>diskSize</DiskSize>
      </Disk>
      <Disk>
        <DiskIndex>diskNum</DiskIndex>
        <DiskSize>diskSize</DiskSize>
      </Disk>
    </Disks>
  </ServerImage>
  <ServerImage name="imageName2">
    <Scope>imageDisclosureScope</Scope>
    <Project>idOfProjWhereImageWasCreated</Project>
    <AvailabilityZone>availZone</AvailabilityZone>
    <Comment>comment</Comment>
    <CreateTime>creationDatetime</CreateTime>
    <Disks>
      <Disk>
        <DiskIndex>diskNum</DiskIndex>
        <DiskSize>diskSize</DiskSize>
      </Disk>
      <Disk>
        <DiskIndex>diskNum</DiskIndex>
        <DiskSize>diskSize</DiskSize>
      </Disk>
    </Disks>
  </ServerImage>
</ServerImages>
```

- If "availability_zone" was specified:
  Only images located in the specified availability zone are displayed.
  Example: If verbose=false, mode=cloning, and disksize=false:
  If some images have availability_zone specified:

```xml
<ServerImages>
  <ServerImage name="imageName1">
    <Scope>imageDisclosureScope</Scope>
    <Project>idOfProjWhereImageWasCreated</Project>
    <AvailabilityZone>availZone</AvailabilityZone>
  </ServerImage>
  <ServerImage name="imageName2">
    <Scope>imageDisclosureScope</Scope>
    <Project>idOfProjWhereImageWasCreated</Project>
    <AvailabilityZone>availZone</AvailabilityZone>
  </ServerImage>
</ServerImages>
```

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>imageName</td>
<td>Name of image</td>
<td>Name of the clone image.</td>
</tr>
</tbody>
</table>
| image DisclosureScope | The scope of disclosure of an image | • public  
  Displayed when the image is public. |
### 1.7.5.17 Change disclosure scope of clone image

#### Description
Changes the disclosure scope of a clone image.

The types of images for which the disclosure scope can be changed are as follows:
- Private images of own project that are disclosed inside the domain.
- Private images that are only disclosed inside the project.

#### Request

**Header**

- Content-Type: application/x-www-form-urlencoded
- Accept: application/xml
- X-Auth-Token: Keystone token (*1)
- X-VA-Project-Id: ID of the project that owns the resource (*2)
X-VA-Domain-Id: ID of the domain that owns the resource (*2)
*1: The scope specified for the token is ignored when this API is executed by the system administrator.
*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

**Method**
PUT

**Path**
/server_images/cloneImageName/move

**Query**
n/a

**Body**

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
</table>
| scope         | • To change the disclosure scope of a clone image to inside the domain Specify "domain".  
                • To change the disclosure scope of a clone image to inside the project Specify "private". |

**Response**

**Status code**
202 Accepted

**Body**

Returns information on tasks executed by each availability zone.
However, <resource id="resourceId"/> is not displayed.

```xml
<tasks>
  <task>
    <id>taskId</id>
    <status>waiting</status>
    <progress>0</progress>
    <starttime>startTime</starttime>
    <AvailabilityZone>availabilityZoneName</AvailabilityZone>
  </task>
  ...
  <id>taskId</id>
</tasks>
```
1.7.5.18 Create snapshot

**Description**

Creates a snapshot from a virtual server.

While snapshots can be created even while the virtual server is running, this may impact on virtual server operations.

Therefore, it is recommended that you stop the virtual server before creating a snapshot.

**Request**

**Header**

- Content-Type: application/x-www-form-urlencoded
- Accept: application/xml
- X-Auth-Token: Keystone token (*1)
- X-VA-Project-Id: ID of the project that owns the resource (*2)
- X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*1: The scope specified for the token is ignored when this API is executed by the system administrator.
*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

**Method**

POST

**Path**

/server_images

**Query**

n/a

**Body**

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>Specify &quot;snapshot&quot;.</td>
</tr>
<tr>
<td>l_server_rid</td>
<td>Specifies the resource ID of the virtual server where the snapshot will be created.</td>
</tr>
</tbody>
</table>

**Response**

**Status code**

202 Accepted

**Body**

Task information

The resource ID of the virtual server is displayed in <resource id="resourceId"/>.
1.7.5.19 Restore snapshot

**Description**
Restores a snapshot for a virtual server.

**Request**

**Header**

- X-Auth-Token: Keystone token (*1)
- X-VA-Project-Id: ID of the project that owns the resource (*2)
- X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*1: The scope specified for the token is ignored when this API is executed by the system administrator.

*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

**Method**
PUT

**Path**
/server_images/snapshotResourceId/restore

**Query**
n/a

**Body**
n/a

**Response**

**Status code**
202 Accepted

**Body**
Task information

1.7.5.20 Delete snapshot

**Description**
Deletes a snapshot.

**Request**

**Header**

- Accept: application/xml
X-Auth-Token: Keystone token (*1)
X-VA-Project-Id: ID of the project that owns the resource (*2)
X-VA-Domain-Id: ID of the domain that owns the resource (*2)
*1: The scope specified for the token is ignored when this API is executed by the system administrator.
*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

Method
DELETE

Path
/server_images/snapshotResourceId

Query

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>Specify &quot;snapshot&quot;.</td>
</tr>
</tbody>
</table>

Body
n/a

Response

Status code
202 Accepted

Body
Task information

1.7.5.21 Retrieve task details

Description
Retrieves detailed information about the specified task.

Request

Header
Accept: application/xml
X-Auth-Token: Keystone token

Method
>GET

Path
/tasks/taskId
Query
n/a

Body
n/a

Response

Status code
200 OK

Body

```xml
<task>
  <id>taskId</id>
  <status>status</status>
  <progress>progressStatus</progress>
  <starttime>startTime</starttime>
  <endtime>endTime</endtime>
  <error>
    <code>msgId</code>
    <message>msgBody</message>
    <arg>arg1</arg>
    <arg>arg2</arg>
    ...
    <cause>
      <product>relatedProdName</product>
      <resource_name>linkedProdResourceName</resource_name>
      <request>inforRequestedForRelatedProd</request>
      <message>returnInfoOfRelatedProd</message>
    </cause>
  </error>
</task>
```

Note: The error tag is only displayed if an error occurs.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>taskId</td>
<td>ID for identifying the task</td>
<td>ID format</td>
</tr>
<tr>
<td>status</td>
<td>Status of the task</td>
<td>waiting, running, completed, error, cancelled</td>
</tr>
<tr>
<td>progressStatus</td>
<td>Progress status (%) of the task</td>
<td>Number from 0 to 100</td>
</tr>
<tr>
<td>startTime</td>
<td>Start time of the task</td>
<td>YYYY-MM-DD hh:mm:ss hh:mm</td>
</tr>
<tr>
<td>endTime</td>
<td>End time of the task</td>
<td>YYYY-MM-DD hh:mm:ss hh:mm</td>
</tr>
</tbody>
</table>

Refer to HTTP response on page 135 for details on the other elements.

1.7.5.22 Enable project

Description

Creates a folder for the specified project, and enables the project. The folder name will be the project ID.
Do not create and delete files concurrently for the same project.

Caution

- The project folder is created in the folder of the domain that it belongs to.
  If the domain folder does not exist, it will be created at the same time.
- The following resource pools are also created in the project folder.
  VADomainImagePool (the disclosure scope is the private image pool in the domain)
  VAPrivateImagePool (the disclosure scope is the private image pool in the project)
  VAProjectNetworkPool (network pool used inside the project)
- If a project folder of the same name already exists, this indicates that it completed normally.
  If the resource pool for the virtual server for SAP does not yet exist, it will be created.

Request

Header

Content-Type: application/x-www-form-urlencoded
Accept: application/xml
X-Auth-Token: Keystone token (*1)
X-VA-Project-Id: ID of the project that owns the resource (*2)
X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*1: Specify the scope of the project to be enabled, and retrieve a token.
*3: Mandatory when this API executed by the system administrator, ignored when executed by other users.

Method

POST

Path

/projects

Query

n/a

Body

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>project_id</td>
<td>Specifies the project ID retrieved from Keystone. Do not include symbols such as hyphens (-). Note that the ID specified here is not case-sensitive (the name of the folder will be converted to lowercase). In case specifying the project the user doesn't belong to, that operation results in error. If you are a system administrator, an error will occur if you specify a value different from the project ID.</td>
</tr>
<tr>
<td>Argument name</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>availability_zone</td>
<td>Specifies the name of the availability zone where the project folder should</td>
</tr>
<tr>
<td></td>
<td>be created. If omitted, project folders will be created in all availability</td>
</tr>
<tr>
<td></td>
<td>zones.</td>
</tr>
</tbody>
</table>

**Response**

**Status code**

202 Accepted

**Body**

Returns information on tasks executed by each availability zone. However, `<resource id="resourceId"/>` is not displayed.

```xml
<tasks>
  <task>
    <id>taskId</id>
    <status>waiting</status>
    <progress>0</progress>
    <starttime>startTime</starttime>
    <AvailabilityZone>availabilityZoneName</AvailabilityZone>
  </task>
  ...
</tasks>
```

1.7.5.23 Disable project

**Description**

Deletes the folder for the specified project, and disables the project. All resources in the project must be deleted before performing this operation. In case specifying the project the user doesn't belong to, that operation results in error. After a project is disabled, even if it is enabled again later, its resources cannot be restored.

**Note**

- The following resource pools in the specified project folder are also deleted.
  - VADomainImagePool (the disclosure scope is the private image pool in the domain)
  - VAPrivateImagePool (the disclosure scope is the private image pool in the project)
  - VANetworkPool (network pool used inside the project)
- The project folder will not be deleted if it contains a resource pool for bare metal services.
- When the system administrator uses this API, an error will occur if a value different from the project ID is specified in the header.

**Request**
Header

Accept: application/xml
X-Auth-Token: Keystone token (*1)
X-VA-Project-Id: ID of the project that owns the resource (*2)
X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*1: Specify the scope of the project to be disabled, and retrieve a token.
When the system administrator uses this API, the scope specified for the token is ignored.
*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

Method
DELETE

Path
/projects/projId

Query

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>availability_zone</td>
<td>Specifies the name of the availability zone where the project folder should be deleted.</td>
</tr>
<tr>
<td></td>
<td>If omitted, project folders will be deleted from all availability zones.</td>
</tr>
</tbody>
</table>

Body

n/a

Response

Status code
202 Accepted

Body

Returns information on tasks executed by each availability zone.
However, <resource id="resourceId"/> is not displayed.

```xml
<tasks>
  <task>
    <id>taskId</id>
    <status>waiting</status>
    <progress>0</progress>
    <starttime>startTime</starttime>
    <AvailabilityZone>availabilityZoneName</AvailabilityZone>
  </task>
  ...
</tasks>
```
1.7.5.24 Retrieve project details

**Description**
Retrieves detailed information about the specified project
When you select the project that you do not belong to, it becomes an error.

**Note**
When the system administrator uses this API, an error will occur if a value different from the project ID is specified in the header.

**Request**

**Header**
- Accept: application/xml
- X-Auth-Token: Keystone token (*1)
- X-VA-Project-Id: ID of the project that owns the resource (*2)
- X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*1: Specify the scope of the project for which detailed information is to be retrieved, and retrieve a token.

When the system administrator uses this API, the scope specified for the token is ignored.

*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

**Method**
GET

**Path**
/projects/projId

**Query**
- n/a

**Body**
- n/a

**Response**

**Status code**
200 OK

**Body**

```xml
<Project name="projFolderName">
  <Domain>domainFolderName</Domain>
  <AvailabilityZones>
    <AvailabilityZone>availZoneWithProjFolder</AvailabilityZone>
    <AvailabilityZone>availZoneWithProjFolder</AvailabilityZone>
    ...
  </AvailabilityZones>
</Project>
```
1.7.5.25 Create network resource

**Description**

Creates a network resource.

A job LAN network is created for virtual server for SAP or for bare metal services, based on the network already created using OpenStack.

The network resource name is generated automatically, and the following prefix is assigned according to the service:

- `VAServiceLAN_****`: Job LAN resources of the virtual server for SAP
- `BAServiceLAN_****`: Job LAN resources for bare metal services

**Caution**

When custom roles are used, network resources are to be created by users who have been assigned the `vmware_network_create` and `net_get_subnet` roles.

**Request**

**Header**

- `Content-Type: application/xml`
- `Accept: application/xml`
- `X-Auth-Token: Keystone token (*1)`
- `X-VA-Project-Id: Project ID of resource creation destination (*2)`
- `X-VA-Domain-Id: Domain ID of resource creation destination (*2)`

*1: The scope specified for the token is ignored when this API is executed by the system administrator.
**Method**
POST

**Path**
/Networks

**Query**
n/a

**Body**

```xml
<Network>
  <Subnet id="subnetIdCreatedUsingOpenstack" />
  <AvailabilityZone>availZone</AvailabilityZone>
</Network>
```

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>subnetIdCreatedUsingOpenstack</td>
<td>Subnet ID created using Openstack (required)</td>
<td>UUID</td>
</tr>
<tr>
<td>Availability zone</td>
<td>Availability zone of the creation destination (optional)</td>
<td>Availability zone name</td>
</tr>
</tbody>
</table>

**Response**

**Status code**
202 Accepted

**Body**
Task information

### 1.7.5.26 Delete network resource

**Description**
Deletes a network resource.

**Request**

**Header**

Accept: application/xml
X-Auth-Token: Keystone token (*1)
X-VA-Project-Id: ID of the project that owns the resource (*2)
X-VA-Domain-Id: ID of the domain that owns the resource (*2)

*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.
*1: The scope specified for the token is ignored when this API is executed by the system administrator.
*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

Method
DELETE

Path
/networks/networkResourceId

Query
n/a

Body
n/a

Response

Status code
202 Accepted

Body
Task information

1.7.5.27 List network resources

Description
Lists the network resources of all availability zones.
Only the network resources owned by the project are displayed.
• When a list of networks for bare metal services (endpoint) is retrieved, the following is displayed.
  • Job LAN network of bare metal services
• When a list of networks for virtual server for SAP (endpoint) is retrieved, the following is displayed.
  • Job LAN network of virtual server for SAP

Request

Header
Accept: application/xml
X-Auth-Token: Keystone token (*1)
X-VA-Project-Id: ID of the project that owns the resource (*2)
X-VA-Domain-Id: ID of the domain that owns the resource (*2)
*1: The scope specified for the token is ignored when this API is executed by the system administrator.
*2: Mandatory when this API executed by the system administrator, ignored when executed by other users.

Method
GET

Path
/networks

Query

<table>
<thead>
<tr>
<th>Argument name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>verbose</td>
<td>To retrieve only some of the information, specify &quot;false&quot;, or to retrieve all of the information, specify &quot;true&quot;. If omitted, &quot;false&quot; is used.</td>
</tr>
</tbody>
</table>

Body
n/a

Response

Status code
200 OK

Body

If verbose=false:

```
<Networks>
  <Network name="networkResourceName1" id="networkResourceId1">
    <AddressSet name="addrSetName1" subnet="subnetAddr" mask="netMask"
      start="startIpAddr" end="endIpAddr">
      <DefaultGateway address="ipAddr" />
    </AddressSet>
    <AvailabilityZone>availZoneName</AvailabilityZone>
  </Network>

  <Network name="networkResourceName2" id="networkResourceId2">
    Omitted
  </Network>
</Networks>
```

If verbose=true:

```
<Networks>
  <Network name="networkResourceName1" id="networkResourceId1">
    <AddressSet name="addrSetName1" subnet="subnetAddr" mask="netMask"
      start="startIpAddr" end="endIpAddr">
      <AddressRange start="startIpAddr" end="endIpAddr" />
      <Exclude>
        <AddressRange start="startIpAddr" end="endIpAddr" />
      </Exclude>
      <Reserve>
        ipAddrAlreadyAssigned (IP addresses are separated with commas)
      </Reserve>
      <DefaultGateway address="ipAddr" />
    </AddressSet>
    <AvailabilityZone>availZoneName</AvailabilityZone>
  </Network>
```

```
Tag name: Network

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of network resource</td>
<td>Name of the network resource.</td>
</tr>
<tr>
<td>id</td>
<td>Network resource ID</td>
<td>Network resource ID.</td>
</tr>
</tbody>
</table>

The network resource names have the following prefixes:
- VAServiceLAN_****: Job LAN resources of the virtual server for SAP
- BAServiceLAN_****: Job LAN resources for bare metal services

Tag name: AddressSet

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of address set resource</td>
<td>Name of the address set resource.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The address set resource names have the following prefixes:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- VAServiceAS_****: Job LAN address set resources of the virtual server for SAP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- BAServiceLAN_****: Job LAN address set resources for bare metal services</td>
</tr>
<tr>
<td>subnet</td>
<td>Subnet address</td>
<td>Subnet address. (example: 192.168.10.0)</td>
</tr>
<tr>
<td>mask</td>
<td>Subnet mask</td>
<td>Subnet mask. (example: 255.255.255.0)</td>
</tr>
<tr>
<td>start</td>
<td>Start IP address</td>
<td>Start IP address for automatic setup. (example: 192.168.10.1)</td>
</tr>
<tr>
<td>end</td>
<td>End IP address</td>
<td>End IP address for automatic setup. (example: 192.168.10.254)</td>
</tr>
</tbody>
</table>

Tag name: Exclude

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address range for manual setup</td>
<td>If the address range for manual setup has not been set, the AddressRange tag will not be displayed. If multiple address ranges for manual setup have been set, multiple AddressRange tags will be displayed.</td>
<td></td>
</tr>
</tbody>
</table>

Tag name: AddressRange

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>start</td>
<td>Start IP address</td>
<td>Start IP address for manual setup. (example: 192.168.10.1)</td>
</tr>
<tr>
<td>end</td>
<td>End IP address</td>
<td>End IP address for manual setup. (example: 192.168.10.1)</td>
</tr>
</tbody>
</table>
Tag name: Reserve

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP address that</td>
<td>IP addresses that have already been assigned.</td>
<td>Example of multiple IP addresses that have already been assigned: 192.168.10.30, 192.168.10.31</td>
</tr>
<tr>
<td>has already been assigned</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Tag name: DefaultGateway

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>address</td>
<td>IP address of the default gateway</td>
<td>IP address of the default gateway. (example: 192.168.10.1)</td>
</tr>
</tbody>
</table>

1.7.6 Status codes in use

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200: OK</td>
<td>OK The request was successful, and the response was returned with the requested information.</td>
</tr>
<tr>
<td></td>
<td>If the page is displayed correctly in the browser, this status code will be returned in most cases.</td>
</tr>
<tr>
<td>201: Created</td>
<td>Created The request was successful, and the URI of the created resource is returned.</td>
</tr>
<tr>
<td></td>
<td>Example: When a request to create a resource using the PUT method is made, this code is returned on completion of that request.</td>
</tr>
<tr>
<td>202: Accepted</td>
<td>Accepted The request was accepted, however, processing has not been completed.</td>
</tr>
<tr>
<td></td>
<td>Example: When a request to create a resource using the PUT method is made, this code is returned when the server has accepted the request but creation of the resource has not been completed. For batch processing.</td>
</tr>
<tr>
<td>400: Bad Request</td>
<td>The request is invalid.</td>
</tr>
<tr>
<td></td>
<td>This code is returned when a client request is abnormal (such as using an undefined method).</td>
</tr>
<tr>
<td>401: Unauthorized</td>
<td>Authentication is required.</td>
</tr>
<tr>
<td></td>
<td>Used when performing Basic or Digest authentication. Most browsers display an authentication dialog when this status is returned.</td>
</tr>
<tr>
<td>403: Forbidden</td>
<td>Forbidden. Access to the resource was denied.</td>
</tr>
<tr>
<td></td>
<td>This code is returned in cases such as the user not having access privileges, or the host received a banned request.</td>
</tr>
<tr>
<td></td>
<td>Example: An attempt is made to access a page from outside the company when access to that page is only allowed internally (intranet).</td>
</tr>
<tr>
<td>404: Not Found</td>
<td>Not found. The resource could not be found.</td>
</tr>
<tr>
<td></td>
<td>This code is also used when the user simply does not have access privileges.</td>
</tr>
<tr>
<td>Status code</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------</td>
</tr>
<tr>
<td>405: Method Not Allowed</td>
<td>Method not allowed. An attempt was made to use a method that is not permitted. Example: When the POST method is used at a location where use of the POST method is not permitted.</td>
</tr>
<tr>
<td>406: Not Acceptable</td>
<td>Not acceptable. This code is returned in cases when unacceptable content is included in an Accept-related header. Example: The server can only accept English or Japanese, however, only zh (Chinese) is included in the Accept-Language: header of the request. Example: The server tried to send an application/pdf document, however, application/pdf was not included in the Accept: header of the request. Example: The server tried to send text in UTF-8 format, however, utf-8 was not included in the Accept-Charset: header of the request.</td>
</tr>
<tr>
<td>408: Request Timeout</td>
<td>Request timeout This code is returned when a request is not completed within a specified time.</td>
</tr>
<tr>
<td>409: Conflict</td>
<td>Conflict The request cannot be completed due to a conflict with an existing resource.</td>
</tr>
<tr>
<td>410: Gone</td>
<td>Gone The file was moved permanently. Its location is unknown. This code is similar to 404 Not Found, however, it indicates that the file will not be recovered.</td>
</tr>
<tr>
<td>412: Precondition Failed</td>
<td>Precondition failed This code is returned when a precondition is false. Example: When an update occurs after the time when the If-Unmodified-Since: header of the request was written.</td>
</tr>
<tr>
<td>413: Request Entity Too Large</td>
<td>Request entity is too large. This code is returned when a request entity exceeds the range permitted by the server. Example: When an attempt was made to send data that exceeded the upper limit of the uploader.</td>
</tr>
<tr>
<td>414: Request-URI Too Long</td>
<td>Request URI is too long. This code is returned when the server rejects processing due to the URI being too long. Example: When an attempt is made to send large data such as image data using the GET method, and the URI is tens of KB in length (the upper limit depends on the server).</td>
</tr>
<tr>
<td>415: Unsupported Media Type</td>
<td>Unsupported media type. This code is returned when the specified media type is not supported.</td>
</tr>
<tr>
<td>416: Requested Range Not Satisfiable</td>
<td>Requested range is not satisfiable. Data exceeding the actual file size was requested. For example, when an attempt is made to retrieve 1025 Bytes, yet the size of the resource is only 1024 Bytes.</td>
</tr>
<tr>
<td>Status code</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| 417: Expectation Failed | Expectation failed  
The extension cannot be responded to, or the proxy server deems that the next server to receive a request cannot respond.  
As a specific example, when an unusual value other than 100-continue is entered in the Expect: header, or the server cannot handle 100 Continue in the first place. |
| 500: Internal Server Error | Internal server error. This code is returned when an internal error occurs on the server.  
For example, when there is a syntax error or setting error in CGI. The syntax of CGI is incorrect. |

### 1.7.7 Error messages

The error message list only describes errors that occur in the delivery platform region manager.

#### Common

<table>
<thead>
<tr>
<th>Status code</th>
<th>Error messages</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>404</td>
<td>${0}: not supported</td>
<td>An unsupported URL was executed.</td>
</tr>
</tbody>
</table>
| 412         | The specified system configuration file, "${0}", is invalid.  
${0}: File path | The definition file does not exist.                                   |
|             |                                                                               | The definition file format is incorrect.                              |
| 404         | Selectable Availability Zone not found. (${0})  
${0}: Reason why availability zone of request transfer destination could not be found | The availability zone of the request transfer destination could not be found. |
| 409         | communication error. {0}  
${0}: Request header name | A required request header was not specified.                           |
<p>| 409         | Connection refused - Connection refused                                       | The delivery platform environment has stopped.                       |
| 409         | certificate verify failed                                                      | The certificate of the delivery platform is invalid.                  |
| 409         | Text explaining why communication was not possible                           | Unable to communicate with the delivery platform environment.         |</p>
<table>
<thead>
<tr>
<th>Status code</th>
<th>Error messages</th>
<th>Cause</th>
</tr>
</thead>
</table>
| 412         | ${0}:invalid format  
${0}: Parameter name | An invalid value was specified for the request parameter.             |
| 400         | ${0} is required  
${0}: Parameter name | A required parameter was not specified.                               |
| 409         | configuration error. target=${0}  
${0}: Region name | The availability zone is not defined on the delivery platform region manager. |
| 403         | ${0}:user not privileged  
${0}: User ID | An error occurred during approval processing.                         |
| 500         | Internal Server Error                               | An error occurred.                                                   |

**Create virtual server**

<table>
<thead>
<tr>
<th>Status code</th>
<th>Error messages</th>
<th>Cause</th>
</tr>
</thead>
</table>
| 404         | ${0}:not found  
${0}: Image name | The specified clone image does not exist.                             |
| 409         | ${0}:duplicate resource name found  
${0}: Virtual server name | A virtual server with the same name as the specified virtual server already exists. |
| 404         | ${0}:not found.(${1})  
${0}: Network endpoint,  
${1}: Literal text "specified resource (network) not found | The specified network resource was not found.                         |

**Change virtual server**

<table>
<thead>
<tr>
<th>Status code</th>
<th>Error messages</th>
<th>Cause</th>
</tr>
</thead>
</table>
| 412         | ${0}:invalid format  
${0}: TemplateLink,name | The VM pool corresponding to the virtual server template is not defined. |
| 404         | ${0}:not found  
${0}: LServer (resource ID of the virtual server) | The specified virtual server does not exist.                          |
| 409         | ${0}:duplicate resource name found  
${0}: Virtual server name | A virtual server with the same name as the new virtual server name already exists. |
### Attach NIC to virtual server

<table>
<thead>
<tr>
<th>Status code</th>
<th>Error messages</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>404</td>
<td>${0}: not found. (${1})&lt;br&gt;${0}: Network endpoint,&lt;br&gt;${1}: Literal text &quot;specified resource (network) not found</td>
<td>The specified network resource was not found</td>
</tr>
</tbody>
</table>

### Create image

<table>
<thead>
<tr>
<th>Status code</th>
<th>Error messages</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>409</td>
<td>${0}: configuration for ${1} not found&lt;br&gt;${0}: Project folder, ${1}: collecting image</td>
<td>The image pool does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>${0}: configuration for ${1} not found&lt;br&gt;${0}: Region, ${1}: Collecting image</td>
<td>The storage pool for images does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>${0}: duplicate resource name found&lt;br&gt;${0}: Clone image name</td>
<td>An image with the same name as the one specified for the clone image already exists.</td>
</tr>
</tbody>
</table>

### Create network resource

<table>
<thead>
<tr>
<th>Status code</th>
<th>Error messages</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>404</td>
<td>${0}: not found. (${1})&lt;br&gt;${0}: Network endpoint,&lt;br&gt;${1}: Can be any of the following: 1. ${1}=availZone:specified subnet uuid is not found&lt;br&gt;2. ${1}=availZone:subnet is for KVM. uuid:subnetUuid&lt;br&gt;3. ${1}=availZone:endpoint does not authorize user&lt;br&gt;4. ${1}=availZone:'keystone endpoint', endpoint(endpoint) is not valid value&lt;br&gt;5. ${1}=availZone:response from endpoint(endpoint) may be not keystone's one&lt;br&gt;6. ${1}=availZone:availability_zone: &lt;availability_zone&gt; not found&lt;br&gt;7. ${1}=availZone:network with specified subnet uuid is not found</td>
<td>Contact Fujitsu technical support if any codes other than the above are displayed. 1. The specified subnet does not exist in availZone or, Custom roles are used but the net_get_subnet role has not been set. 2. subnetUuid is not the subnet for virtual server for SAP. 3. The specified token is not suitable. 4. The endpoint of the specified token is invalid. 5. The endpoint of the specified token</td>
</tr>
<tr>
<td>Status code</td>
<td>Error messages</td>
<td>Cause</td>
</tr>
<tr>
<td>-------------</td>
<td>--------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>is other than keystone.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6. The specified &lt;availability_zone&gt; does not exist</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7. The network to which the specified subnet belongs does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>network create failed.(${0})</td>
<td>1. Network resources have already been created for the specified subnet.</td>
</tr>
<tr>
<td></td>
<td>${0}: Can be any of the following:</td>
<td>2. Processing is in progress elsewhere for the subnet.</td>
</tr>
<tr>
<td></td>
<td>1. already exist network linked for specified subnet. (subnetUuid)</td>
<td>That is, network resources are being created for the subnet, or the subnet is being deleted in OpenStack.</td>
</tr>
<tr>
<td></td>
<td>2. specified subnet(subnetUuid) is being used by other operations.</td>
<td>3. The subnet uuid is incorrect or is not the one created for the virtual server for SAP.</td>
</tr>
<tr>
<td></td>
<td>3. not for linked subnet, resource_id: subnetUuid</td>
<td>Contact Fujitsu technical support if any codes other than the above are displayed.</td>
</tr>
</tbody>
</table>

Delete network resource

<table>
<thead>
<tr>
<th>Status code</th>
<th>Error messages</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>404</td>
<td>${0}: not found.(${1})</td>
<td>The specified network resource was not found</td>
</tr>
<tr>
<td></td>
<td>${0}: Network endpoint,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>${1}: Literal text &quot;specified resource (network) not found</td>
<td></td>
</tr>
</tbody>
</table>

Enable project/Disable project/Retrieve project details

<table>
<thead>
<tr>
<th>Status code</th>
<th>Error messages</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>409</td>
<td>The value of ${0}, ${1}, is invalid.</td>
<td>This error occurs only if the API was executed by the administrator.</td>
</tr>
<tr>
<td></td>
<td>${0}: project_id, ${1}: value specified for project_id</td>
<td>The project ID specified in the header is different from the one specified for the project_id argument.</td>
</tr>
</tbody>
</table>
1.8 Other (common)

1.8.1 Generate URLs when using APIs

For URLs to be used by APIs of the following categories, use URLs of the "compute" type from the Service catalog retrieved from the identity service.

- Key pairs
- Server console output
- Images with size attribute
- Server password

The endpoint URL is returned in the following format by the identity service.

```
https://hostName/v2/{tenant_id}
```

Create the URL in one of the following formats:

- If you remember the tenant_id:
  Join the path name of each API in the host section of the end point URL
- If you do not remember the tenant_id:
  The endpoint URL and the path name of each API with "v2/tenant_id" removed from the beginning

For URLs to be used by APIs of the following categories, use URLs of the "image" type from the Service catalog retrieved from the identity service.

- Image Service API v2 Images

The endpoint URL is returned in the following format by the identity service.

```
https://hostName
```

Join the path name of each API in the host section of the end point URL, and create the URL.

1.8.2 API list

### Key pairs

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /v2/tenant_id/os-keypairs</td>
<td>Retrieves a list of the key pairs associated with the account</td>
</tr>
<tr>
<td>2</td>
<td>POST /v2/tenant_id/os-keypairs</td>
<td>Creates or imports key pairs</td>
</tr>
<tr>
<td>Item</td>
<td>API</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>-------------</td>
</tr>
<tr>
<td>Create or import key pair</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>DELETE /v2/{tenant_id}/os-keypairs/{keypair_name}</td>
<td>Deletes a key pair</td>
</tr>
<tr>
<td>Delete key pair</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>GET /v2/{tenant_id}/os-keypairs/{keypair_name}</td>
<td>Displays the key pairs associated with the account</td>
</tr>
<tr>
<td>Show key pair information</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Server console output**

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>POST /v2/{tenant_id}/servers/{server_id}/action</td>
<td>Retrieves the console output of the server instance</td>
</tr>
<tr>
<td>Get console output for an instance</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Images with size attribute**

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /v2/{tenant_id}/images/detail</td>
<td>Lists details of available images and their sizes</td>
</tr>
<tr>
<td>List details of images</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>GET /v2/{tenant_id}/images/{image_id}</td>
<td>Retrieves the details of the specified image and its size</td>
</tr>
<tr>
<td>Get image details</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Server password**

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /v2/{tenant_id}/servers/{server_id}/os-server-password</td>
<td>Retrieves the management password of the specified server</td>
</tr>
<tr>
<td>Get server password</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Image Service API v2 Images**

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PATCH /v2/images/{image_id}</td>
<td>Updates the specified image</td>
</tr>
<tr>
<td>Update an image</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>GET /v2/images/?limit,marker,name,visibility,member_status,owner,status,size_min,size_max,sort_key,sort_dir,tag</td>
<td>Lists public virtual server images</td>
</tr>
<tr>
<td>List images</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Item</td>
<td>API</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>-------------</td>
</tr>
<tr>
<td>3</td>
<td>GET /v2/images{/limit,marker,name,visibility, member_status,owner,status,size_min,size_max, sort_key,sort_dir,tag}</td>
<td>List images</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Searches public virtual server images by name, and display a list</td>
</tr>
<tr>
<td>4</td>
<td>GET /v2/images{/image_id}</td>
<td>Get an image</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Retrieves the details of the specified image</td>
</tr>
<tr>
<td>5</td>
<td>DELETE /v2/images{/image_id}</td>
<td>Delete an image</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Deletes the specified image</td>
</tr>
</tbody>
</table>

**Members**

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>POST /v2/images{/image_id}/members</td>
<td>Create image member</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Adds a tenant (project) for sharing images</td>
</tr>
<tr>
<td>2</td>
<td>GET /v2/images{/image_id}/members</td>
<td>List image members</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Displays a list of tenants (projects) that are sharing images</td>
</tr>
<tr>
<td>3</td>
<td>GET /v2/images{/image_id}/members{/member_id}</td>
<td>Show image member details</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Retrieves details of members who are sharing images</td>
</tr>
<tr>
<td>4</td>
<td>DELETE /v2/images{/image_id}/members{/member_id}</td>
<td>Delete image member</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Deletes the specified tenant (project) from an image sharing member</td>
</tr>
<tr>
<td>5</td>
<td>PUT /v2/images{/image_id}/members{/member_id}</td>
<td>Update image member</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Changes a tenant (project) for sharing images</td>
</tr>
</tbody>
</table>

**1.8.3 Request header**

<table>
<thead>
<tr>
<th>Header</th>
<th>Specified value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accept</td>
<td>application/json</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>X-Auth-Token</td>
<td>Authentication token</td>
</tr>
</tbody>
</table>

**1.8.4 API error codes**

<table>
<thead>
<tr>
<th>Error code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>500, other codes possible</td>
<td>Server Error, cloudServersFault</td>
</tr>
<tr>
<td>Error code</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>400</td>
<td>badRequest</td>
</tr>
<tr>
<td>401</td>
<td>unauthorized</td>
</tr>
<tr>
<td>403</td>
<td>Forbidden, resizeNotAllowed</td>
</tr>
<tr>
<td>404</td>
<td>itemNotFound</td>
</tr>
<tr>
<td>405</td>
<td>badMethod</td>
</tr>
<tr>
<td>409</td>
<td>conflictingRequest</td>
</tr>
<tr>
<td>413</td>
<td>overLimit</td>
</tr>
<tr>
<td>415</td>
<td>badMediaType, Unsupported Media Type</td>
</tr>
<tr>
<td>501</td>
<td>notImplemented</td>
</tr>
<tr>
<td>503</td>
<td>serviceUnavailable</td>
</tr>
</tbody>
</table>

### 1.8.5 Notes

- In some cases, an API (key pair list, etc.) used to display a list of resources may return only part of the availability zone information. If this happens, it is assumed that infrastructure maintenance is in progress, so wait for a few moments (at least one minute) and then execute the API again.
- Description of API error code
  - This code is returned when execution of the API below is not approved under the operator privileges (role).
  - GET /v2/images/{image_id}/members/{member_id}

### 1.8.6 API details

#### 1.8.6.1 List key pairs

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/os-keypairs</td>
<td>Lists key pairs associated with the account.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the list key pairs request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.
Response

Example. List key pairs: JSON response

```json
{
    "keypairs": [
        {
            "keypair": {
                "name": "keypair-601a2305-4f25-41ed-89c6-2a966fc8027a",
                "public_key": "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAAAgQC+Eo/R2Rn+gaGTkFs7I62ZjsI079Kk1KbMi8F+K1TD4bVQHHn+kV+4gRgkCGkCDBdQDqGfpADPs877DYX9n46FrAA24FES8TNkhatifpn9NdQYWA+ikU8Cuv1EKGufKPki/k7JLos/gHi2hy7QUwgtRvcfVD/vgQZ0Vw/mGR9Q== Generated byNova\n"
            }
        }
    ]
}
```

1.8.6.2 Create or import key pair

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td><code>/v2/{tenant_id}/os-keypairs</code></td>
<td>Creates or imports a key pair.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

⚠️ Before deleting an account (user), the key pair created by it must be deleted.

Caution

Request

This table shows the URI parameters for the create or import keypair request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the create or import keypair request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Name to associate with the keypair.</td>
</tr>
<tr>
<td>public_key</td>
<td>string (Optional)</td>
<td>Public ssh key to import. If not provided, a key is generated.</td>
</tr>
<tr>
<td>availability_zone</td>
<td>string (Optional)</td>
<td>Availability zone for the key pair. If omitted, the availability zone will be determined automatically from the UUID of the domain that the request execution user belongs to.</td>
</tr>
</tbody>
</table>
**Example. Create or import key pair: JSON request**

```json
{
  "keypair": {
    "name": "keypair-dab428fe-6186-4a14-b3de-92131f76cd39",
    "public_key": "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAAGDx8nkQv/zgGqB4rMYmIf+6A4l6Rr+o/61HBQdW5aYd44bd8JttDCE/F/pNMr0I\nRE+PiqSPO8nPbHw0010JeMH9gYgmnFyY3/OcJ02RhIPyyxYpyv9FrY+y\n2YiUkpwFOcLImyrxEsYxpD/0d3ac30bNH6Sw9JD9UZHYcpSxsIbECHw== Generated by\nNova",
    "availability_zone": "jp-east-1a"
  }
}
```

**Response**

**Example. Create or import key pair: JSON response**

```json
{
  "keypair": {
    "fingerprint": "1e:2c:9b:56:79:4b:45:77:ca:7a:98:2c:b0:d5:3c",
    "name": "keypair-dab428fe-6186-4a14-b3de-92131f76cd39",
    "public_key": "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAAGDx8nkQv/zgGqB4rMYmIf+6A4l6Rr+o/61HBQdW5aYd44bd8JttDCE/F/pNMr0I\nRE+PiqSPO8nPbHw0010JeMH9gYgmnFyY3/OcJ02RhIPyyxYpyv9FrY+y\n2YiUkpwFOcLImyrxEsYxpD/0d3ac30bNH6Sw9JD9UZHYcpSxsIbECHw== Generated by\nNova",
    "user_id": "fake"
  }
}
```

### 1.8.6.3 Delete key pair

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DELETE</td>
<td>/v2/{tenant_id}/os-keypairs/{keypair_name}?{availability_zone}</td>
<td>Deletes a key pair.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

**Request**

This table shows the URI parameters for the delete key pair request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{keypair_name}</td>
<td>String</td>
<td>The name associated with the key pair.</td>
</tr>
<tr>
<td>availability_zone</td>
<td>String (optional)</td>
<td>Availability zone for the key pair.</td>
</tr>
</tbody>
</table>

This operation does not accept a request body and does not return a response body.
1.8.6.4 Show key pair information

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/os-keypairs/ {keypair_name}?availability_zone</td>
<td>Retrieves a key pair associated with the account.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the delete key pair request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{keypair_name}</td>
<td>String</td>
<td>The name associated with the key pair.</td>
</tr>
<tr>
<td>availability_zone</td>
<td>String (Optional)</td>
<td>Availability zone for the key pair.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. Show key pair information: JSON response

```
{
  "keypair": {
    "public_key": "ssh-rsa AAAAB3NzaC1yc2EAAAADAQAB AAQABgQDCS1LxfzqB+e5yHdUSXvb xKkaajj1fuHv+GArdseqPjfkKJ6no5echpin7dJp0FLMNJKxJZE3WIIRu25CQrJntmi7no27RkDF AGaTFbje30WY4A1HLKeKAB5tFhwYQ1r7CjMdoC/DY7UvKaatwLhH4Wvh5vMmXgF7AFzdkI28urwo +Q== nova@use03147k5-eth0\n",
    "name": "hpdefault",
    "availability_zone":"jp-east-1a"
  }
}
```

1.8.6.5 Get console output for an instance

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/servers/{server_id}/action</td>
<td>Gets console output for a server instance.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the get console output for an instance request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>
Name | Type | Description
---|---|---
{server_id} | UUID | The UUID for the server of interest to you.

This table shows the body parameters for the et console output for an instance request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
</table>
| length | string | Number of lines to fetch from end of console log.

**Example. Get console output: JSON request**

```json
{
  "os-getConsoleOutput": {
    "length": 50
  }
}
```

**Response**

**Example. Get console output: JSON response**

```json
{
  "output": "FAKE CONSOLE OUTPUT
ANOTHER
LAST LINE"
}
```

1.8.6.6 List details for images

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/images/detail</td>
<td>Lists details for available images. Includes the image size.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the list details for images request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. List details for images: JSON response**

```json
{
  "images": [
    {
      "OS-EXT-IMG-SIZE:size": "74185822",
```
"created": "2011-01-01T01:02:03Z",
"id": "70a599e0-31e7-49b7-b260-868f441e862b",
"links": [
  {
    "href": "http://openstack.example.com/v2/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
    "rel": "self"
  },
  {
    "href": "http://openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
    "rel": "bookmark"
  },
  {
    "href": "http://glance.openstack.example.com/openstack/images/70a599e0-31e7-49b7-b260-868f441e862b",
    "rel": "alternate",
    "type": "application/vnd.openstack.image"
  }
],
"metadata": {
  "architecture": "x86_64",
  "auto_disk_config": "True",
  "kernel_id": "nokernel",
  "ramdisk_id": "nokernel"
},
"minDisk": 0,
"minRam": 0,
"name": "fakeimage7",
"progress": 100,
"status": "ACTIVE",
"updated": "2011-01-01T01:02:03Z"
},
{
  "OS-EXT-IMG-SIZE:size": "25165824",
  "created": "2011-01-01T01:02:03Z",
  "id": "155d900f-4e14-4e4c-a73d-069cbf4541e6",
  "links": [
    {
      "href": "http://openstack.example.com/v2/openstack/images/155d900f-4e14-4e4c-a73d-069cbf4541e6",
      "rel": "self"
    },
    {
      "href": "http://openstack.example.com/openstack/images/155d900f-4e14-4e4c-a73d-069cbf4541e6",
      "rel": "bookmark"
    },
    {
      "href": "http://glance.openstack.example.com/openstack/images/155d900f-4e14-4e4c-a73d-069cbf4541e6",
      "rel": "alternate",
      "type": "application/vnd.openstack.image"
    }
  ],
  "metadata": {
    "architecture": "x86_64",
    "kernel_id": "nokernel",
    "ramdisk_id": "nokernel"
  },
  "minDisk": 0,
  "minRam": 0,
  "name": "fakeimage123456",
  "progress": 100,
  "status": "ACTIVE",
  "updated": "2011-01-01T01:02:03Z"
}
}
"OS-EXT-IMG-SIZE:size": "26360814",
"created": "2011-01-01T01:02:03Z",
"id": "c905cedb-7281-47e4-8a62-f26bc5fc4c77",
"links": [ 
  { "href": "http://openstack.example.com/v2/openstack/images/c905cedb-7281-47e4-8a62-f26bc5fc4c77",
    "rel": "self"
  },
  { "href": "http://openstack.example.com/openstack/images/c905cedb-7281-47e4-8a62-f26bc5fc4c77",
    "rel": "bookmark"
  },
  { "href": "http://glance.openstack.example.com/openstack/images/c905cedb-7281-47e4-8a62-f26bc5fc4c77",
    "rel": "alternate",
    "type": "application/vnd.openstack.image"
  }
],
"metadata": {
  "kernel_id": "155d900f-4e14-4e4c-a73d-069cbf4541e6",
  "ramdisk_id": "null",
  "minDisk": 0,
  "minRam": 0,
  "name": "fakeimage123456",
  "progress": 100,
  "status": "ACTIVE",
  "updated": "2011-01-01T01:02:03Z"
},

"OS-EXT-IMG-SIZE:size": "84035174",
"created": "2011-01-01T01:02:03Z",
"id": "cedef40a-ed67-4d10-800e-17455edce175",
"links": [ 
  { "href": "http://openstack.example.com/v2/openstack/images/cedef40a-ed67-4d10-800e-17455edce175",
    "rel": "self"
  },
  { "href": "http://openstack.example.com/openstack/images/cedef40a-ed67-4d10-800e-17455edce175",
    "rel": "bookmark"
  },
  { "href": "http://glance.openstack.example.com/openstack/images/cedef40a-ed67-4d10-800e-17455edce175",
    "rel": "alternate",
    "type": "application/vnd.openstack.image"
  }
],
"metadata": {
  "kernel_id": "nokernel",
  "ramdisk_id": "nokernel",
  "minDisk": 0,
  "minRam": 0,
  "name": "fakeimage123456",
  "progress": 100,
  "status": "ACTIVE",
  "updated": "2011-01-01T01:02:03Z"
},

"OS-EXT-IMG-SIZE:size": "83594576",
"created": "2011-01-01T01:02:03Z"
1.8.6.7 Get image details

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/images/{image_id}</td>
<td>Gets details for a specified image. Includes the image size.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the get image details request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The ID for the tenant or account in a multi-tenancy cloud.</td>
</tr>
<tr>
<td>{image_id}</td>
<td>String</td>
<td>Image ID stored through the image API. Typically a UUID.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. Get image details: JSON response

```json
{
   "id": "76fa36fc-c930-4bf3-8c8a-ea2a2420deb6",
   "links": [
      {
         "href": "http://openstack.example.com/v2/openstack/images/76fa36fc-c930-4bf3-8c8a-ea2a2420deb6",
         "rel": "self"
      },
      {
         "href": "http://openstack.example.com/openstack/images/76fa36fc-c930-4bf3-8c8a-ea2a2420deb6",
         "rel": "bookmark"
      },
      {
         "href": "http://glance.openstack.example.com/openstack/images/76fa36fc-c930-4bf3-8c8a-ea2a2420deb6",
         "rel": "alternate",
         "type": "application/vnd.openstack.image"
      }
   ],
   "metadata": {
      "kernel_id": "nokernel",
      "ramdisk_id": "nokernel"
   },
   "minDisk": 0,
   "minRam": 0,
   "name": "fakeimage123456",
   "progress": 100,
   "status": "ACTIVE",
   "updated": "2011-01-01T01:02:03Z"
}
```
1.8.6.8 Get server password

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td><code>/v2/servers/{server_id}/os-server-password</code></td>
<td>Gets the administrative password for a specified server.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the get server password request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{server_id}</td>
<td>UUID</td>
<td>The UUID for the server of interest to you.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. Get server password: JSON response**

```json
{}
```
1.8.6.9 Update image

PATCH /v2/images/{image_id}

Normal response codes: 200

It is necessary to specify application/openstack-images-v2.1-json-patch for the Content-Type of the request header.

e7db3b45-8db7-47ad-8109-3fb55c2c24fd as an example:

```
[{
  "op": "replace", "path": "/name", "value": "Fedora 17"},
  {"op": "replace", "path": "/tags", "value": ["fedora", "beefy"]}
]
```

The response body shows the updated image entity. For example:

```
{
  "id": "e7db3b45-8db7-47ad-8109-3fb55c2c24fd",
  "name": "Fedora 17",
  "status": "queued",
  "visibility": "public",
  "tags": ["fedora", "beefy"],
  "created_at": "2012-08-11T17:15:52Z",
  "updated_at": "2012-08-11T17:15:52Z",
  "self": "/v2/images/e7db3b45-8db7-47ad-8109-3fb55c2c24fd",
  "file": "/v2/images/e7db3b45-8db7-47ad-8109-3fb55c2c24fd/file",
  "schema": "/v2/schemas/image"
}
```

The PATCH method can also be used to add or remove image properties. To add a custom user-defined property such as "login-user" to an image, use the following example request.

```
[{
  "op": "add", "path": "/login-user", "value": "kvothe"}
]
```

Similarly, to remove a property such as "login-user" from an image, use the following example request.

```
[{
  "op": "remove", "path": "/login-user"
}
]
```

See Appendix B for more details about the 'application/openstack-images-v2.1-json-patch' media type.
Property protections

Version 2.2 of the Images API acknowledges the ability of a cloud provider to employ property protections. Thus, there may be image properties that may not be updated or deleted by non-admin users.

1.8.6.10 List images

GET /v2/images

Normal response codes: 200

Request body ignored.

Response body will be a list of images available to the client. For example:

```json
{
  "images": [
    {
      "id": "da3b75d9-3f4a-40e7-8a2c-bfab23927dea",
      "name": "cirros-0.3.0-x86_64-uec-ramdisk",
      "status": "active",
      "visibility": "public",
      "size": 2254249,
      "checksum": "2cec138d7da2aa59038ef8c9aec2390",
      "tags": ["ping", "pong"],
      "created_at": "2012-08-10T19:23:50Z",
      "updated_at": "2012-08-10T19:23:50Z",
      "self": "/v2/images/da3b75d9-3f4a-40e7-8a2c-bfab23927dea",
      "file": "/v2/images/da3b75d9-3f4a-40e7-8a2c-bfab23927dea/file",
      "schema": "/v2/schemas/image"
    },
    {
      "id": "0d5bcbc7-b066-4217-83f4-71116a60a399a",
      "name": "cirros-0.3.0-x86_64-uec",
      "status": "active",
      "visibility": "public",
      "size": 25165824,
      "checksum": "2f81976cae15c16ef0010c51e3a6c163",
      "tags": [],
      "created_at": "2012-08-10T19:23:50Z",
      "updated_at": "2012-08-10T19:23:50Z",
      "self": "/v2/images/0d5bcbc7-b066-4217-83f4-71116a60a399a",
      "file": "/v2/images/0d5bcbc7-b066-4217-83f4-71116a60a399a/file",
      "schema": "/v2/schemas/image"
    },
    {
      "id": "e6421c88-b1ed-4407-8824-b57298249091",
      "name": "cirros-0.3.0-x86_64-uec-kernel",
      "status": "active",
      "visibility": "public",
      "size": 4731440,
      "checksum": "cfb203e7267a28e435dbcb05af5910a9",
      "tags": [],
      "created_at": "2012-08-10T19:23:49Z",
      "updated_at": "2012-08-10T19:23:49Z",
      "self": "/v2/images/e6421c88-b1ed-4407-8824-b57298249091",
      "file": "/v2/images/e6421c88-b1ed-4407-8824-b57298249091/file",
      "schema": "/v2/schemas/image"
    }
  ],
  "first": "/v2/images?limit=3",
  "next": "/v2/images?limit=3&marker=e6421c88-b1ed-4407-8824-b57298249091",
  "schema": "/v2/schemas/images"
}
```
Pagination

This call is designed to return a subset of the larger collection of images while providing a link that can be used to retrieve the next. You should always check for the presence of a 'next' link and use it as the URI in a subsequent HTTP POST request. You should follow this pattern until there is a 'next' link is no longer provided. The next link will preserve any query parameters you send in your initial request. The 'first' link can be used to jump back to the first page of the collection.

If you prefer to paginate through images manually, the API provides two query parameters: 'limit' and 'marker'. The limit parameter is used to request a specific page size.

Expect a response to a limited request to return between zero and limit items. The marker parameter is used to indicate the id of the last-seen image. The typical pattern of limit and marker is to make an initial limited request then to use the id of the last image from the response as the marker parameter in a subsequent limited request.

Filtering

The list operation accepts several types of query parameters intended to filter the results of the returned collection.

A client can provide direct comparison filters using most image attributes (i.e. name=Ubuntu, visibility=public, etc). A client cannot filter on tags or anything defined as a 'link' in the json-schema (i.e. self, file, schema).

The 'size_min' and 'size_max' query parameters can be used to do greater-than and less-than filtering of images based on their 'size' attribute ('size' is measured in bytes and refers to the size of an image when stored on disk). For example, sending a size_min filter of 1048576 and size_max of 4194304 would filter the container to include only images that are between one and four megabytes in size.

Sorting

The results of this operation can be ordered using the 'sort_key' and 'sort_dir' parameters.

The API uses the natural sorting of whatever image attribute is provided as the 'sort_key'.

All image attributes can be used as the sort_key (except tags and link attributes).

The sort_dir parameter indicates in which direction to sort. Acceptable values are 'asc' (ascending) and 'desc' (descending). Defaults values for sort_key and sort_dir are 'created_at' and 'desc'.

Property Protections

Version 2.2 of the Images API acknowledges the ability of a cloud provider to employ property protections. Thus, there may be image properties that will not appear in the list images response for non-admin users.

1.8.6.11 Get an Image

GET /v2/images/<IMAGE_ID>

Normal response codes: 200

Request body ignored.
### Response parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>csapi:UUID</td>
<td>The UUID of the image.</td>
</tr>
<tr>
<td>name</td>
<td>xsd:string</td>
<td>The name of the image. Value might be null (JSON null data type).</td>
</tr>
<tr>
<td>status</td>
<td>xsd:string</td>
<td>The image status.</td>
</tr>
<tr>
<td>visibility</td>
<td>xsd:string</td>
<td>The image visibility. A valid value is public or private. Default is private.</td>
</tr>
<tr>
<td>size</td>
<td>xsd:int</td>
<td>The size of the image data, in bytes. The value might be null (JSON null data type).</td>
</tr>
<tr>
<td>checksum</td>
<td>xsd:String</td>
<td>Hash that is used over the image data. The Image service uses this value for verification. The value might be null (JSON null data type).</td>
</tr>
<tr>
<td>tags</td>
<td>xsd:list</td>
<td>A list of tag objects.</td>
</tr>
<tr>
<td>created_at</td>
<td>xsd:String</td>
<td>The date and time when the resource was created.</td>
</tr>
<tr>
<td>updated_at</td>
<td>xsd:String</td>
<td>The date and time when the resource was updated.</td>
</tr>
<tr>
<td>self</td>
<td>xsd:String</td>
<td>The URL for the virtual machine image.</td>
</tr>
<tr>
<td>file</td>
<td>xsd:String</td>
<td>The URL for the virtual machine image file.</td>
</tr>
<tr>
<td>schema</td>
<td>xsd:String</td>
<td>The URL for schema of the virtual machine image.</td>
</tr>
<tr>
<td>owner</td>
<td>csapi:UUID</td>
<td>The ID of the owner, or tenant, of the image. The value might be null (JSON null data type).</td>
</tr>
<tr>
<td>auth_version</td>
<td>xsd:String</td>
<td>Authentication format for accessing data stored in an image. The authentication formats that are displayed are &quot;v3&quot; or &quot;v2.0&quot;. Only when the displayed authentication format is &quot;v3&quot;, and &quot;owner&quot; is your project, it is possible to change the information for accessing image data set in the image by following <em>2.1.1.8.6.18 Update an image (v1)</em>.</td>
</tr>
</tbody>
</table>

Response body is a single image entity. Using GET /v2/image/da3b75d9-3f4a-40e7-8a2c-bfab23927dea as an example:

```json
{
    "id": "da3b75d9-3f4a-40e7-8a2c-bfab23927dea",
    "name": "cirros-0.3.0-x86_64-uec-ramdisk",
    "status": "active",
    "visibility": "public",
    "size": 2254249,
    "checksum": "2cec138d7dae2aa59038ef8c9aec2390",
    "tags": ["ping", "pong"],
    "created_at": "2012-08-10T19:23:50Z",
    "updated_at": "2012-08-10T19:23:50Z",
    "self": "/v2/images/da3b75d9-3f4a-40e7-8a2c-bfab23927dea",
    "file": "/v2/images/da3b75d9-3f4a-40e7-8a2c-bfab23927dea/file",
    "schema": "/v2/schemas/image"
}
```
Property Protections
Version 2.2 of the Images API acknowledges the ability of a cloud provider to employ property protections. Thus, there may be some image properties that will not appear in the image detail response for non-admin users.

1.8.6.12 Delete image
DELETE /v2/images/{image_id}
Normal response codes: 204

Caution
If the password of the user who registered (imported) the image was changed, image deletion will fail. In such a case, convey the image UUID to the operator and request deletion of the image.

1.8.6.13 Create image member
POST /v2/images/{image_id}/members
Normal response codes: 200

Preconditions
• The specified image must exist.
• You can only add a member to an image which visibility attribute is private.
• You must be the owner of the specified image.

Synchronous Postconditions
• With correct permissions, you can see the member status of the image as pending through API calls.

Troubleshooting
• Even if you have correct permissions, if the visibility attribute is set to public, the request returns the HTTP 403 error code. Ensure that you meet the preconditions and run the request again. If the request fails again, review your API request.
• If the specified member is already a member for the image, the service returns the HTTP 409 Conflict error code. In case you meant a different member, double-check that you specified the correct member.

Implement the workflow below when sharing images for use.
1. Add a member to the image that will be shared.
   After the image provider creates an image with the visibility attribute set to "private", add a member (project ID/tenant ID) who will share the image.
   To add a member to the image, use "Create image member".
2. Start using the shared image.
   The status of the member who the image was shared with changes to "accepted" for using the image.
   To change the member status of the image, use "Update image member".
3. By changing the member status of the image to "accepted", the shared image will be displayed in the image list retrieved using "List images".

To stop sharing images, perform the following procedure:
1. Stop using shared images.
The image provider deletes the member for whom usage is to be stopped from the image being shared.

To delete a member from an image, use "Delete image member".

By deleting a member of an image, the member will no longer be able to view or use the image being shared.

Caution

Even if the status of a member with whom an image is being shared is changed to "rejected", that member is still able to use the image, so it is necessary to request the image provider to delete the member.

Request

This table shows the URI parameters for the create image member request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>image_id</td>
<td>uuid</td>
<td>Image ID stored through the image API. Typically a UUID.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the create image member request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>member</td>
<td>string</td>
<td>Image member ID. For example, the tenant ID of the user with whom the image is being shared.</td>
</tr>
</tbody>
</table>

Example. Create image member: JSON request

```json
{
  "member": "8989447062e04a818baf9e073fd04fa7"
}
```

Response

```json
{
  "created_at": "2013-09-20T19:22:19Z",
  "image_id": "a96be11e-8536-4910-92cb-de50aa19dfe6",
  "member_id": "8989447062e04a818baf9e073fd04fa7",
  "schema": "/v2/schemas/member",
  "status": "pending",
  "updated_at": "2013-09-20T19:25:31Z"
}
```

1.8.6.14 List image member

GET /v2/images/<IMAGE_ID>/members

Normal response codes: 200

If a user with whom this image is shared makes this call, the member list contains only information for that user.

If a user with whom this image has not been shared makes this call, the call returns the HTTP 404 status code.
Preconditions

- The specified image must exist.
- You must be the owner or a member of the specified image.

Request

This table shows the URI parameters for the list image member request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>image_id</td>
<td>uuid</td>
<td>Image ID stored through the image API. Typically a UUID.</td>
</tr>
</tbody>
</table>

This operation does not accept a request body.

Response

```
{
   "members": [
      {
         "created_at": "2013-10-07T17:58:03Z",
         "image_id": "dbc999e3-c52f-4200-bedd-3b18fe7f87fe",
         "member_id": "123456789",
         "schema": "/v2/schemas/member",
         "status": "pending",
         "updated_at": "2013-10-07T17:58:03Z"
      },
      {
         "created_at": "2013-10-07T17:58:55Z",
         "image_id": "dbc999e3-c52f-4200-bedd-3b18fe7f87fe",
         "member_id": "987654321",
         "schema": "/v2/schemas/member",
         "status": "accepted",
         "updated_at": "2013-10-08T12:08:55Z"
      }
   ],
   "schema": "/v2/schemas/members"
}
```

1.8.6.15 Show image member details

GET /v2/images/<IMAGE_ID>/members/<member_id>

Normal response codes: 200

Preconditions

- The specified image must exist.
- You must be the owner or a member of the specified image.

Request

This table shows the URI parameters for the show image member details request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>image_id</td>
<td>uuid</td>
<td>Image ID stored through the image API. Typically a UUID.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>member_id</td>
<td>string</td>
<td>Image member ID. For example, the tenant ID of the user with whom the image is being shared.</td>
</tr>
</tbody>
</table>

This operation does not accept a request body.

**Response**

```json
{
    "status": "pending",
    "created_at": "2013-11-26T07:21:21Z",
    "updated_at": "2013-11-26T07:21:21Z",
    "image_id": "0ae74cc5-5147-4239-9ce2-b0c580f7067e",
    "member_id": "8989447062e04a818baf9e073fd04fa7",
    "schema": "/v2/schemas/member"
}
```

### 1.8.6.16 Delete image member

**DELETE /v2/images/<IMAGE_ID>/members/<member_id>**

Normal response codes: 204

**Preconditions**

- The specified image must exist.
- You must be the owner of the specified image.

**Synchronous Postconditions**

- The specified member is removed from the image members.

**Troubleshooting**

- Even if you have correct permissions, if you are not the owner of the specified image, the request returns the HTTP 403 error code. Ensure that you meet the preconditions and run the request again. If the request fails again, review your API request.

**Request**

This table shows the URI parameters for the delete image member request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>image_id</td>
<td>uuid</td>
<td>Image ID stored through the image API. Typically a UUID.</td>
</tr>
<tr>
<td>member_id</td>
<td>string</td>
<td>Image member ID. For example, the tenant ID of the user with whom the image is being shared.</td>
</tr>
</tbody>
</table>

This operation does not accept a request body and does not return a response body.

### 1.8.6.17 Update image member

**PUT /v2/images/<IMAGE_ID>/members/<member_id>**

Normal response codes: 200

**Preconditions**
• The specified images must exist.
• You must be a member of the specified image.

Synchronous Postconditions
• If you update the member status to accepted and have the correct permissions, you see the image in list images responses.
• With correct permissions, you can see the updated member status of the image through API calls.

Request
This table shows the URI parameters for the update image member request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>image_id</td>
<td>uuid</td>
<td>Image ID stored through the image API. Typically a UUID.</td>
</tr>
<tr>
<td>member_id</td>
<td>string</td>
<td>Image member ID. For example, the tenant ID of the user with whom the image is being shared.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the update image member request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>status</td>
<td>string</td>
<td>The status of this image member.</td>
</tr>
</tbody>
</table>

Example. Update image member: JSON request

```json
{
    "status": "accepted"
}
```

Response

```json
{
    "created_at": "2013-09-20T19:22:19Z",
    "image_id": "a96be11e-8536-4910-92cb-de50aa19dfe6",
    "member_id": "8989447062e04a818baf9e073fd04fa7",
    "schema": "/v2/schemas/member",
    "status": "accepted",
    "updated_at": "2013-09-20T20:15:31Z"
}
```

1.8.6.18 Update image (v1)

**PUT /v1/images/{image_id}**

Changes information required to access image data.

The information that can be changed is: domain name, user name and password.

When registering images using the virtual server import service, the domain name, user name and password of the user who registered the image are set as image information. When the password of the user who registered an image is changed, use this API to change the password set for the image.
If the password set for an image is not changed, an error may occur when creating an instance, creating a volume with the image specified, or deleting the image. An error will not occur during the period while cache of the volume or image exists.

If the domain name and user name are changed, the password must be changed as well.

**Notes when using this API**

When *Get an Image* on page 196 is used to retrieve image data, and the authentication format displayed in "auth_version" is "v3" and the project ID displayed in "owner" is your own project, this API can be used to change the information set for the image. The information cannot be changed in other cases.

Normal response codes: 200

**Request**

Specify the following parameters in the HTTP header.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x-image-meta-domain</td>
<td>String</td>
<td>Specify the new domain name (if changing the domain name).</td>
</tr>
<tr>
<td>x-image-meta-user</td>
<td>String</td>
<td>Specify the new user name (if changing the user name).</td>
</tr>
</tbody>
</table>
| x-image-meta-password| string    | Specify the new password to (if changing the password). Encode special characters such as *@* and *@*.
                                      *
                                      *@* must be encoded as "%40", and *:* as
                                      "%3A".                                      |

Specify the x-image-meta-domain, x-image-meta-user, and x-image-meta-password:

- Specify at least one HTTP header. Multiple HTTP headers can also be specified.

⚠️ The information can be changed only if the image status is "active".

**Response**

```json
{
  "image": {
    "status": "active",
    "deleted": false,
    "container_format": "bare",
    "min_ram": 0,
    "updated_at": "2016-04-26T04:57:19",
    "owner": "496c27733a4141eb824e4cb9932b8372",
    "min_disk": 0,
    "is_public": false,
    "deleted_at": null,
    "id": "da3b75d9-3f4a-40e7-8a2c-bfab23927dea",
    "size": 13167616,
    "virtual_size": null,
    "name": "cirros-0.3.0-x86_64",
    "checksum": null,
    "created_at": "2016-04-26T04:44:35",
    "disk_format": "raw",
    "properties": {},
    "protected": false
  }
}
Part 2: Storage

Topics:
- System storage
- Extended storage
- Backup
- Snapshot
- Images
- Extended storage replication
- Object storage
2.1 System storage

2.1.1 Generate URLs when using APIs

For URLs to be used by APIs of the following categories, use URLs of the "blockstoragev2" type from the Service catalog retrieved from the identity service.

- volume types
- volume
- Block Storage API v2 Volumes
- Upload volume to image service as image

For URLs to be used by APIs of the following categories, use URLs of the "compute" type from the Service catalog retrieved from the identity service.

- Volume extension

The endpoint URL is returned in the following format by the identity service.

```
https://hostName/v2/{tenant_id}
```

Create the URL in one of the following formats:

- If you remember the tenant_id:
  Join the path name of each API in the host section of the end point URL
- If you do not remember the tenant_id:
  The endpoint URL and the path name of each API with "/v2/{tenant_id}" removed from the beginning

2.1.2 API list

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /v2/{tenant_id}/types</td>
<td>Retrieves a list of volume types</td>
</tr>
<tr>
<td></td>
<td>List volume types</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>GET /v2/{tenant_id}/types/{volume_type_id}</td>
<td>Retrieves information related to the specified volume type</td>
</tr>
<tr>
<td></td>
<td>Show volume type information</td>
<td></td>
</tr>
</tbody>
</table>
### Volume

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /v2/{tenant_id}/volumes</td>
<td>List volumes</td>
</tr>
<tr>
<td></td>
<td>GET /v2/{tenant_id}/volumes/detail</td>
<td>List volumes (detailed)</td>
</tr>
<tr>
<td>2</td>
<td>GET /v2/{tenant_id}/volumes/{volume_id}</td>
<td>Show volume information</td>
</tr>
<tr>
<td>3</td>
<td>DELETE /v2/{tenant_id}/volumes/{volume_id}</td>
<td>Delete volume</td>
</tr>
</tbody>
</table>

### Volume extension

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /v1.1/{tenant_id}/os-volumes</td>
<td>List volumes</td>
</tr>
<tr>
<td></td>
<td>GET /v1.1/{tenant_id}/os-volumes/detail</td>
<td>List details for volumes</td>
</tr>
<tr>
<td>2</td>
<td>POST /v1.1/{tenant_id}/os-volumes</td>
<td>Create volume</td>
</tr>
<tr>
<td>3</td>
<td>GET /v1.1/{tenant_id}/os-volumes/{volume_id}</td>
<td>Show volume information</td>
</tr>
<tr>
<td>4</td>
<td>DELETE /v1.1/{tenant_id}/os-volumes/{volume_id}</td>
<td>Delete volume</td>
</tr>
<tr>
<td>5</td>
<td>POST /v1.1/{tenant_id}/os-snapshots</td>
<td>Create snapshot</td>
</tr>
<tr>
<td>6</td>
<td>GET /v1.1/{tenant_id}/os-snapshots</td>
<td>List snapshots</td>
</tr>
<tr>
<td>7</td>
<td>GET /v1.1/{tenant_id}/os-snapshots/detail</td>
<td>List details for snapshots</td>
</tr>
<tr>
<td>8</td>
<td>GET /v1.1/{tenant_id}/os-snapshots/{snapshot_id}</td>
<td>Show snapshot</td>
</tr>
<tr>
<td>9</td>
<td>DELETE /v1.1/{tenant_id}/os-snapshots/{snapshot_id}</td>
<td>Delete snapshot</td>
</tr>
</tbody>
</table>
### Block Storage API v2 Volumes

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | POST /v2/{tenant_id}/volumes  
Create volume | Creates a volume |
| 2    | PUT /v2/{tenant_id}/volumes/{volume_id}  
Update volume | Updates a volume |

### Upload volume to image service as image

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | POST /v2/{tenant_id}/volumes/{volume_id}/action  
Create image | Creates an image from a volume |

### Extend volume size

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | POST /v2/{tenant_id}/volumes/{volume_id}/action  
Extend size of volume | Extends the size of a volume |

### 2.1.3 Request header

<table>
<thead>
<tr>
<th>Header</th>
<th>Specified value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accept</td>
<td>application/json</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>X-Auth-Token</td>
<td>Authentication token</td>
</tr>
</tbody>
</table>

### 2.1.4 API error codes

<table>
<thead>
<tr>
<th>Error code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>500, other codes possible</td>
<td>Server Error</td>
</tr>
<tr>
<td>400</td>
<td>badRequest</td>
</tr>
<tr>
<td>401</td>
<td>unauthorized</td>
</tr>
<tr>
<td>403</td>
<td>Forbidden, resizeNotAllowed</td>
</tr>
<tr>
<td>404</td>
<td>itemNotFound</td>
</tr>
</tbody>
</table>
2.1.5 Notes

When an API (volume list, etc.) used to display a list of resources is executed, only some of the availability zone information may be returned. If this happens, it is assumed that infrastructure maintenance is in progress, so wait for a few moments (at least one minute) and then execute the API again.

2.1.6 API details

2.1.6.1 List volume types

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/types</td>
<td>Lists volume types.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the list volume types request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. List volume types: JSON response

```
{
  "volume_types": [
    {
      "extra_specs": {
        "capabilities": "gpu"
      },
      "id": "6685584b-1eac-4da6-b5c3-555430cf68ff",
      "name": "SSD"
    },
    {
      "extra_specs": {},
      "id": "8eb69a46-df97-4e41-9586-9a40a7533803",
      "name": "SATA"
    }
  ]
}
```

2.1.6.2 Show volume type information

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/types/{volume_type_id}</td>
<td>Shows information about a specified volume type.</td>
</tr>
</tbody>
</table>
Normal response codes: 200

Request
This table shows the URI parameters for the show volume type information request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
<tr>
<td>{volume_type_id}</td>
<td>UUID</td>
<td>The unique identifier for an existing volume type.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response
Example. Show volume type information: JSON response

```json
{
  "volume_type": {
    "id": "6685584b-1eac-4da6-b5c3-555430cf68ff",
    "name": "SSD",
    "extra_specs": {
      "capabilities": "gpu"
    },
    "availability_zone": "jp-east-1a"
  }
}
```

2.1.6.3 List volumes

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/volumes</td>
<td>Lists summary information for all Block Storage volumes that the tenant who submits the request can access.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request
This table shows the URI parameters for the list volumes request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response
Example. List volumes: JSON response

```json
{
  "volumes": [
    {
      "id": "45baf976-c20a-4894-a7c3-c94b7376bf55",
      "links": [
```
2.1.6.4 List volumes (detailed)

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/volumes/detail</td>
<td>Lists detailed information for all Block Storage volumes that the tenant who submits the request can access.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the list volumes (detailed) request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. List volumes (detailed): JSON response**

```json
{
    "volumes": [
        {
            "status": "available",
            "attachments": []
        }
    ]
}
```
2.1.6.5 Show volume information

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/volumes/{volume_id}</td>
<td>Shows information about a specified volume.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the show volume information request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
<tr>
<td>{volume_id}</td>
<td>UUID</td>
<td>The unique identifier of an existing volume.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

This table shows the Response parameters:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>volume</td>
<td>dict</td>
<td>A volume object.</td>
</tr>
<tr>
<td>attachments</td>
<td>list</td>
<td>Instance attachment information.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If this volume is attached to a server instance, the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>attachments list includes the UUID of the attached</td>
</tr>
<tr>
<td></td>
<td></td>
<td>server, an attachment UUID, the name of the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>attached host, if any, the volume UUID, the device,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>and the device UUID.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Otherwise, this list is empty.</td>
</tr>
<tr>
<td>links</td>
<td>list</td>
<td>The volume links.</td>
</tr>
<tr>
<td>availability_zone</td>
<td>string</td>
<td>The availability zone.</td>
</tr>
<tr>
<td>os-vol-host-attr:host</td>
<td>string</td>
<td>Current back-end of the volume.</td>
</tr>
<tr>
<td>encrypted</td>
<td>boolean</td>
<td>If true, this volume is encrypted.</td>
</tr>
<tr>
<td>snapshot_id</td>
<td>UUID</td>
<td>The UUID of the source volume snapshot.</td>
</tr>
<tr>
<td>id</td>
<td>UUID</td>
<td>The UUID of the volume.</td>
</tr>
<tr>
<td>size</td>
<td>int</td>
<td>The size of the volume, in gibibytes (GiB).</td>
</tr>
<tr>
<td>user_id</td>
<td>UUID</td>
<td>The UUID of the user.</td>
</tr>
<tr>
<td>os-vol-tenant-attr:tenant_id</td>
<td>UUID</td>
<td>The tenant ID which the volume belongs to.</td>
</tr>
<tr>
<td>os-vol-mig-status-attr:migstat</td>
<td>string</td>
<td>The status of this volume migration (None means</td>
</tr>
<tr>
<td></td>
<td></td>
<td>that a migration is not currently in progress).</td>
</tr>
<tr>
<td>metadata</td>
<td>dict</td>
<td>One or more metadata key and value pairs that are</td>
</tr>
<tr>
<td></td>
<td></td>
<td>associated with the volume.</td>
</tr>
<tr>
<td>status</td>
<td>string</td>
<td>The volume status.</td>
</tr>
<tr>
<td>description</td>
<td>string</td>
<td>The volume description.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>source_volid</td>
<td>UUID</td>
<td>The UUID of the source volume.</td>
</tr>
<tr>
<td>os-vol-mig-status-attr:name_id</td>
<td>UUID</td>
<td>The volume ID that this volume name on the backend is based on.</td>
</tr>
<tr>
<td>name</td>
<td>string</td>
<td>The volume name.</td>
</tr>
<tr>
<td>bootable</td>
<td>boolean</td>
<td>Enables or disables the bootable attribute. You can boot an instance from a bootable volume.</td>
</tr>
<tr>
<td>created_at</td>
<td>dateTime</td>
<td>The date and time when the resource was created. The date and time stamp format is ISO 8601: CCYY-MM-DDThh:mm:ss hh:mm. For example, 2015-08-27T09:49:58-05:00. The hh:mm value, if included, is the time zone as an offset from UTC.</td>
</tr>
<tr>
<td>volume_type</td>
<td>string</td>
<td>The volume type. In an environment with multiple-storage back ends, the scheduler determines where to send the volume based on the volume type. For information about how to use volume types to create multiple-storage back ends, see Configure multiple-storage back ends.</td>
</tr>
<tr>
<td>volume_image_metadata</td>
<td>dict</td>
<td>One or more metadata key and value pairs that are associated with the image of volume.</td>
</tr>
</tbody>
</table>

**Example. Show volume information: JSON response**

```json
{
    "volume": {
        "status": "available",
        "attachments": [],
        "links": [
            {
                "href": "http://localhost:8776/v2/0c2eba2c5af04d3f9e9d0d410b371fde/volumes/5aa119a8-d25b-45a7-8d1b-88e127885635",
                "rel": "self"
            },
            {
                "href": "http://localhost:8776/0c2eba2c5af04d3f9e9d0d410b371fde/volumes/5aa119a8-d25b-45a7-8d1b-88e127885635",
                "rel": "bookmark"
            }
        ],
        "availability_zone": "nova",
        "bootable": "false",
        "os-vol-host-attr:host": "ip-10-168-107-25",
        "source_volid": null,
        "snapshot_id": null,
        "id": "5aa119a8-d25b-45a7-8d1b-88e127885635",
        "description": "Super volume.",
        "name": "vol-002",
        "created_at": "2013-02-25T02:40:21.000000",
        "volume_type": "None",
        "os-vol-tenant-attr:tenant_id": "0c2eba2c5af04d3f9e9d0d410b371fde",
        "size": 1,
        "metadata": {
            "contents": "not junk"
        }
    }
}```
2.1.6.6 Delete volume

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DELETE</td>
<td>/v2/{tenant_id}/volumes/{volume_id}</td>
<td>Deletes a specified volume.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

Request

This table shows the URI parameters for the show volume information request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
<tr>
<td>{volume_id}</td>
<td>UUID</td>
<td>The unique identifier of an existing volume.</td>
</tr>
</tbody>
</table>

This operation does not accept a request body and does not return a response body.

2.1.6.7 List volumes

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v1.1/{tenant_id}/os-volumes</td>
<td>Lists the volumes associated with the account.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the list volumes request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. List volumes: JSON response

```json
{
    "volumes": [
        {
            "id": "521752a6-acf6-4b2d-bc7a-119f9148cd8c",
            "displayName": "vol-001",
            "displayDescription": "Another volume.",
            "size": 30,
            "volumeType": "289da7f8-6440-407c-9fb4-7db01ec49164",
            "metadata": {
                "contents": "junk"
            },
            "availabilityZone": "us-east1",
        }
    ]
}
```
2.1.6.8 List details for volumes

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td><code>/v1.1/{tenant_id}/os-volumes/detail</code></td>
<td>Lists details for a specified volume.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the list details for volumes request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>{tenant_id}</code></td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. List details for volumes: JSON response**

```json
{
  "volumes": [
    {
      "id": "521752a6-acf6-4b2d-bc7a-119f9148cd8c",
      "displayName": "vol-001",
      "displayDescription": "Another volume.",
      "size": 30,
      "volumeType": "289da7f8-6440-407c-9fb4-7db01ec49164",
      "metadata": {
        "contents": "junk"
      },
      "availabilityZone": "us-east1",
      "snapshotId": null,
      "attachments": [],
      "createdAt": "2012-02-14T20:53:07Z"
    },
    {
      "id": "76b8950a-8594-4e5b-8dce-0d9a9c696358",
      "displayName": "vol-002",
      "displayDescription": "Yet another volume.",
      "createdAt": "2012-03-15T19:10:03Z"
    }
  ]
}
```
2.1.6.9 Create volume

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v1.1/{tenant_id}/os-volumes</td>
<td>Creates a volume.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

⚠️ The upper limit for the volume size is 16 TB.

Caution

Request

This table shows the URI parameters for the create volume request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the create volume request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>display_name</td>
<td>string (Optional)</td>
<td>Volume name.</td>
</tr>
<tr>
<td>display_description</td>
<td>string (Optional)</td>
<td>Volume description.</td>
</tr>
<tr>
<td>size</td>
<td>integer</td>
<td>Volume size, in GB.</td>
</tr>
<tr>
<td>volume_type</td>
<td>string (Optional)</td>
<td>Volume type identifier.</td>
</tr>
<tr>
<td>metadata</td>
<td>hash (Optional)</td>
<td>A set of key/value pairs. These pair replace any existing key/value pairs in the resources metadata with matching keys. Any key/value pairs in the parameter with keys that do not occur in the existing resource metadata are added to the resources metadata.</td>
</tr>
<tr>
<td>availability_zone</td>
<td>string (Optional)</td>
<td>Volume availability zone.</td>
</tr>
</tbody>
</table>
Example. Create volume: JSON request

```json
{
    "volume": {
        "display_name": "vol-001",
        "display_description": "Another volume.",
        "size": 30,
        "volume_type": "289da7f8-6440-407c-9fb4-7db01ec49164",
        "metadata": {
            "contents": "junk"
        },
        "availability_zone": "us-east1"
    }
}
```

Response

Example. Create volume: JSON response

```json
{
    "volume": {
        "id": "521752a6-acf6-4b2d-bc7a-119f9148cd8c",
        "displayName": "vol-001",
        "displayDescription": "Another volume.",
        "size": 30,
        "volumeType": "289da7f8-6440-407c-9fb4-7db01ec49164",
        "metadata": {
            "contents": "junk"
        },
        "availabilityZone": "us-east1",
        "snapshotId": null,
        "attachments": [],
        "createdAt": "2012-02-14T20:53:07Z"
    }
}
```

2.1.6.10 Show volume information

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v1.1/{tenant_id}/os-volumes/{volume_id}</td>
<td>Shows information for a specified volume</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the show volume information request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
<tr>
<td>{volume_id}</td>
<td>UUID</td>
<td>The unique identifier for a volume.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.
Response

Example. Show volume information: JSON response

```
{
  "volume": {
    "id": "521752a6-acf6-4b2d-bc7a-119f9148cd8c",
    "displayName": "vol-001",
    "displayDescription": "Another volume.",
    "size": 30,
    "volumeType": "289da7f8-6440-407c-9fb4-7db01ec49164",
    "metadata": {
      "contents": "junk"
    },
    "availabilityZone": "us-east1",
    "snapshotId": null,
    "attachments": [],
    "createdAt": "2012-02-14T20:53:07Z"
  }
}
```

2.1.6.11 Delete volume

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DELETE</td>
<td>/v1.1/{tenant_id}/os-volumes/{volume_id}</td>
<td>Deletes a specified volume.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

Request

This table shows the URI parameters for the delete volume request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
<tr>
<td>{volume_id}</td>
<td>String</td>
<td>The unique identifier for a volume.</td>
</tr>
</tbody>
</table>

This operation does not accept a request body and does not return a response body.

2.1.6.12 Create snapshot

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v1.1/{tenant_id}/os-snapshots</td>
<td>Creates a snapshot.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the create snapshot request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the create snapshot request:
display_name | string | (Optional) A display name for the snapshot.
display_description | string | (Optional) A description of the snapshot.
volume_id | uuid | The source volume for the snapshot.
force | boolean | (Optional) If true the operation will snapshot the source volume even if it is attached to a server. If it is false it will only snapshot if the volume is not attached.

Example. Create snapshot: JSON request

```json
{
    "snapshot": {
        "display_name": "snap-001",
        "display_description": "Daily backup",
        "volume_id": "521752a6-acf6-4b2d-bc7a-119f9148cd8c",
        "force": true
    }
}
```

Response

Example. Create snapshot: JSON response

```json
{
    "snapshot": {
        "id": "3fbbcccf-d058-4502-8844-6feeffdf4cb5",
        "displayName": "snap-001",
        "displayDescription": "Daily backup",
        "volumeId": "521752a6-acf6-4b2d-bc7a-119f9148cd8c",
        "status": "available",
        "size": 30,
        "createdAt": "2012-02-29T03:50:07Z"
    }
}
```

2.1.6.13 List snapshots

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v1.1/{tenant_id}/os-snapshots</td>
<td>Lists snapshots.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the list snapshots request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
</tbody>
</table>
This operation does not require a request body.

Response

Example. List snapshots: JSON response

```json
{
    "snapshots": [
        {
            "id": "3fbbccccf-d058-4502-8844-6fefe6fd4cb5",
            "displayName": "snap-001",
            "displayDescription": "Daily backup",
            "volumeId": "521752a6-acf6-4b2d-bc7a-119f9148cd8c",
            "status": "available",
            "size": 30,
            "createdAt": "2012-02-29T03:50:07Z"
        },
        {
            "id": "e479997c-650b-40a4-9dfe-77655818b0d2",
            "displayName": "snap-002",
            "displayDescription": "Weekly backup",
            "volumeId": "76b8950a-8594-4e5b-8dce-0dfa9c696350",
            "status": "available",
            "size": 25,
            "createdAt": "2012-03-19T01:52:47Z"
        }
    ]
}
```

2.1.6.14 List details for snapshots

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v1.1/{tenant_id}/os-snapshots/detail</td>
<td>Lists details for a specified snapshot.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the list details for snapshots request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. List details for snapshots: JSON response

```json
{
    "snapshots": [
        {
            "id": "3fbbccccf-d058-4502-8844-6fefe6fd4cb5",
            "displayName": "snap-001",
            "displayDescription": "Daily backup",
            "volumeId": "521752a6-acf6-4b2d-bc7a-119f9148cd8c",
            "status": "available",
            "size": 30,
            "createdAt": "2012-02-29T03:50:07Z"
        },
        {
            "id": "e479997c-650b-40a4-9dfe-77655818b0d2",
            "displayName": "snap-002",
            "displayDescription": "Weekly backup",
            "volumeId": "76b8950a-8594-4e5b-8dce-0dfa9c696350",
            "status": "available",
            "size": 25,
            "createdAt": "2012-03-19T01:52:47Z"
        }
    ]
}
```
2.1.6.15 Show snapshot

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v1.1/{tenant_id}/os-snapshots/ {snapshot_id}</td>
<td>Shows information for a specified snapshot.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the show snapshot request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
<tr>
<td>{snapshot_id}</td>
<td>String</td>
<td>The unique identifier for a snapshot.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. Show snapshot: JSON response

```json
{
    "snapshot": {
        "id": "3fbcc5cf-d058-4502-8844-6feffdf4cb5",
        "displayName": "snap-001",
        "displayDescription": "Daily backup",
        "volumeId": "521752a6-acf6-4b2d-bc7a-119f9148cd8c",
        "status": "available",
        "size": 30,
        "createdAt": "2012-02-29T03:50:07Z",
        "availability_zone": "jp-east-1a"
    }
}
```
### 2.1.6.16 Delete snapshot

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DELETE</td>
<td>/v1.1/{tenant_id}/os-snapshots/{snapshot_id}</td>
<td>Deletes a specified snapshot from the account.</td>
</tr>
</tbody>
</table>

This operation is asynchronous. You must list snapshots repeatedly to determine whether the snapshot was deleted.

Normal response codes: 202

**Request**

This table shows the URI parameters for the delete snapshot request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
<tr>
<td>{snapshot_id}</td>
<td>String</td>
<td>The unique identifier for a snapshot.</td>
</tr>
</tbody>
</table>

This operation does not accept a request body and does not return a response body.

---

### 2.1.6.17 Create volume

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/volumes</td>
<td>Creates a volume.</td>
</tr>
</tbody>
</table>

To create a bootable volume, include the ID of the image from which you want to create the volume in the imageRef attribute in the request body.

Normal response codes: 202

**Request**

This table shows the URI parameters for the create volume request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the create volume request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>availability_zone</td>
<td>string (Optional)</td>
<td>The availability zone.</td>
</tr>
<tr>
<td>source_volid</td>
<td>Uuid (Optional)</td>
<td>To create a volume from an existing volume, specify the ID of the existing volume.</td>
</tr>
<tr>
<td>description</td>
<td>string (Optional)</td>
<td>The volume description.</td>
</tr>
<tr>
<td>snapshot_id</td>
<td>Uuid (Optional)</td>
<td>To create a volume from an existing snapshot, specify the ID of the existing volume snapshot.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>size</td>
<td>Int</td>
<td>The size of the volume, in GBs.</td>
</tr>
<tr>
<td></td>
<td>(Optional if snapshot_id is specified)</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>The volume name.</td>
</tr>
<tr>
<td></td>
<td>(Optional)</td>
<td></td>
</tr>
<tr>
<td>imageRef</td>
<td>Uuid</td>
<td>The ID of the image from which you want to create the volume. Required to create a bootable volume.</td>
</tr>
<tr>
<td></td>
<td>(Optional)</td>
<td></td>
</tr>
<tr>
<td>volume_type</td>
<td>string</td>
<td>The associated volume type.</td>
</tr>
<tr>
<td></td>
<td>(Optional)</td>
<td></td>
</tr>
<tr>
<td>metadata</td>
<td>string</td>
<td>One or more metadata key and value pairs to associate with the volume.</td>
</tr>
<tr>
<td></td>
<td>(Optional)</td>
<td></td>
</tr>
</tbody>
</table>

**Example. Create volume: JSON request**

```json
{
    "volume": {
        "availability_zone": null,
        "source_volid": null,
        "description": null,
        "snapshot_id": null,
        "size": 10,
        "name": "my_volume",
        "imageRef": null,
        "volume_type": null,
        "metadata": {}
    }
}
```

**Response**

**Example. Create volume: JSON response**

```json
{
    "volume": {
        "status": "creating",
        "name": "my_volume",
        "attachments": [],
        "availability_zone": "nova",
        "bootable": "false",
        "created_at": "2014-02-21T19:52:04.949734",
        "description": null,
        "volume_type": "M1",
        "snapshot_id": null,
        "source_volid": null,
        "metadata": {},
        "id": "93c2e2aa-7744-4fd6-a31a-80c4726b08d7",
        "size": 10
    }
}
```
This table shows the body parameters for the create volume response:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>status</td>
<td>String (Required)</td>
<td>The volume status.</td>
</tr>
<tr>
<td>name</td>
<td>String (Required)</td>
<td>The volume name.</td>
</tr>
<tr>
<td>attachments</td>
<td>String (Required)</td>
<td>One or more instance attachments.</td>
</tr>
<tr>
<td>availability_zone</td>
<td>String (Required)</td>
<td>The availability zone.</td>
</tr>
<tr>
<td>bootable</td>
<td>Boolean (Required)</td>
<td>Enables or disables the bootable attribute. You can boot an instance from a bootable volume.</td>
</tr>
<tr>
<td>created_at</td>
<td>Datetime (Required)</td>
<td>Date and time when the volume was created.</td>
</tr>
<tr>
<td>description</td>
<td>String (Required)</td>
<td>The volume description.</td>
</tr>
<tr>
<td>volume_type</td>
<td>String (Required)</td>
<td>The associated volume type.</td>
</tr>
<tr>
<td>snapshot_id</td>
<td>Uuid (Required)</td>
<td>To create a volume from an existing volume snapshot, specify the ID of the existing volume snapshot.</td>
</tr>
<tr>
<td>source_volid</td>
<td>Uuid (Required)</td>
<td>To create a volume from an existing volume, specify the ID of the existing volume.</td>
</tr>
<tr>
<td>metadata</td>
<td>String (Required)</td>
<td>One or more metadata key and value pairs to associate with the volume.</td>
</tr>
<tr>
<td>id</td>
<td>Uuid (Required)</td>
<td>The volume ID.</td>
</tr>
<tr>
<td>size</td>
<td>Int (Required)</td>
<td>The size of the volume, in GBs.</td>
</tr>
</tbody>
</table>

**2.1.6.18 Update volume**

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUT</td>
<td>/v2/{tenant_id}/volumes/{volume_id}</td>
<td>Updates a volume.</td>
</tr>
</tbody>
</table>

Normal response codes: 200
Request

This table shows the URI parameters for the update volume request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
<tr>
<td>{volume_id}</td>
<td>UUID</td>
<td>The unique identifier of an existing volume.</td>
</tr>
</tbody>
</table>

This table shows the query parameters for the update volume request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>string</td>
<td>A description of the volume.</td>
</tr>
<tr>
<td>name</td>
<td>string</td>
<td>The name of the volume.</td>
</tr>
</tbody>
</table>

Example. Update volume: JSON request

```json
{
  "volume": {
    "name": "vol-003",
    "description": "This is yet, another volume."
  }
}
```

Response

Example. Update volume: JSON response

```json
{
  "volume": {
    "status": "available",
    "attachments": [],
    "links": [
      {
        "href": "http://localhost:8776/v2/0c2eba2c5af04d3f9e9d0d410b371fde/volumes/5aa119a8-d25b-45a7-8d1b-88e127885635",
        "rel": "self"
      },
      {
        "href": "http://localhost:8776/0c2eba2c5af04d3f9e9d0d410b371fde/volumes/5aa119a8-d25b-45a7-8d1b-88e127885635",
        "rel": "bookmark"
      }
    ],
    "availability_zone": "nova",
    "source_volid": null,
    "snapshot_id": null,
    "id": "5aa119a8-d25b-45a7-8d1b-88e127885635",
    "description": "This is yet, another volume.",
    "name": "vol-003",
    "created_at": "2013-02-25T02:40:21.000000",
    "volume_type": "None",
    "size": 1,
    "metadata": {
      "contents": "not junk"
    }
  }
}
2.1.6.19 Create image

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/volumes/{volume_id}/action</td>
<td>Creates an image.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

Caution: If using Linux in the system volume, before creating an image check whether the following directory exists and whether it is a symbolic link.

/var/lib/cloud/instance

If the above directory is a real directory, then delete it and after creating the image, check the following.

- An instance is created using the image, and personalized processing such as password change is performed.

Also, before creating the image, check the descriptions in the configuration file below. If an external disk is configured to perform automount, a check for a non-existent disk will be performed when starting the server that used the created image, and as a result the server may not start normally.

/etc/fstab

Created images are registered to each availability zone so that they can be used from all of them. If image registration fails for another availability zone, an error is recorded in the properties (sync_status_reason) of images that exist in the availability zone where the image was created. If an error occurs, delete the image and then create it again.

Example:

"sync_status_reason":"ERROR:A timeout occurred during synchronize image the server(http://1.1.1.1:9292) in the image operation. image_id=681995bb-26c3-421e-9734-0c02b1acbf7d"

Before using an image, check the following for the image that is returned with the image_id of the response information:

- The image exists
- The status of the image is active
- The sync_status_reason property does not exist for the image

If using Windows in the system volume, perform the following tasks before and after creating an image:

Refer to 「the FUJITSU Cloud Service K5 IaaS Service Specification」 for details on the tasks.

- Tasks before creating an image
  Perform tasks 2 to 6 on the Windows operating system.
1. Retrieve a snapshot
   Use the snapshot feature to retrieve a snapshot of the system storage.
   Refer to "Create snapshot on page 240" for details on how to create a snapshot.

2. Allow remote access
   Remote access is allowed by default. If remote access is currently set to not allowed,
   use the following procedure to allow remote access.
   
   Windows Server 2012 SE
   Click Start > Control Panel > System and Security > Allow remote access, and in
   the dialog box, select Allow remote connections.
   
   Windows Server 2008 SE/EE
   Click Start > Control Panel > System and Security > Allow remote access, and in
   the dialog box, select Allow connections from computers running any version
   of Remote Desktop or Allow connections only from computers running Remote
   Desktop with Network Level Authentication.

3. Edit the sysprep response file
   Edit the sysprep response file if necessary. The sysprep response file is stored in the
   following location on the virtual server:
   
   Windows Server 2012 SE
   C:\Windows\System32\Syspreplans_w2k12.xml
   
   Windows Server 2008 SE
   C:\Windows\System32\Syspreplans_w2k8_se.xml
   
   Windows Server 2008 EE
   C:\Windows\System32\Syspreplans_w2k8_ee.xml

4. Delete log files
   Delete the following log files described in the definition file on the virtual server.
   Cloudbase-init log files.
   The paths of the Cloudbase-init log files are described in the following locations on
   Windows.
   
   • cloudbase-init-unattend log file
     
     cloudbase-initInstallLocation\conf\cloudbase-init-unattend.conf
     
     Log directory is specified in: logdir setting in the [DEFAULT] section
     
     Log file is specified in: logfile setting in the [DEFAULT] section
     
     The normal log file is:
     
     C:\Program Files (x86)\Cloudbase Solutions\Cloudbase-Init\log\cloudbase-init-
     unattend.log
   
   • cloudbase-init log file
     
     cloudbase-initInstallLocation\conf\cloudbase-init.conf
     
     Log directory is specified in: logdir setting in the [DEFAULT] section
     
     Log file is specified in: logfile setting in the [DEFAULT] section
     
     The normal log file is:
     
     C:\Program Files (x86)\Cloudbase Solutions\Cloudbase-Init\log\cloudbase-init.log

5. Delete registry information
   Delete registry information of Cloudbase-init.
Windows operating system, 32-bit version
HKEY_LOCAL_MACHINE\Software\Cloudbase Solutions\Cloudbase-Init
Windows operating system, 64-bit version
HKEY_LOCAL_MACHINE\Software\Wow6432Node\Cloudbase Solutions\Cloudbase-Init

6. Execute sysprep

Execute the following batch file:
C:\Windows\System32\sysprep\vsysprep.bat

- Tasks after creating an image
  1. Restore the snapshot
     Use the snapshot feature to restore the snapshot of the system storage.
     Refer to "Restore volume from the snapshot on page 242" for details.
  2. Start the virtual server
     Start the virtual server.
     Refer to "Start server on page 95" for details.

---

**Request**

This table shows the URI parameters for the create image request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
<tr>
<td>{volume_id}</td>
<td>UUID</td>
<td>The unique identifier of an existing volume.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the create image request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>container_format</td>
<td>string (Optional)</td>
<td>Specifies the container format of the image to be created. Specify &quot;bare&quot;.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If omitted, &quot;bare&quot; will be used.</td>
</tr>
<tr>
<td>disk_format</td>
<td>string (Optional)</td>
<td>Specifies the disk format of the image to be created. Specify &quot;raw&quot;.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If omitted, &quot;raw&quot; will be used.</td>
</tr>
<tr>
<td>image_name</td>
<td>string</td>
<td>Specifies the name of the image to be created.</td>
</tr>
<tr>
<td>force</td>
<td>boolean (Optional)</td>
<td>If &quot;False&quot; is specified, images cannot be created when a volume is mounted on the instance. If &quot;True&quot; is specified, images can be created even when a volume is mounted on the instance. If you will specify this parameter, stop the operating system beforehand. If omitted, &quot;false&quot; will be used.</td>
</tr>
</tbody>
</table>
Example. Create volume: JSON request

```json
{
  "os-volume_upload_image": {
    "container_format": "bare",
    "disk_format": "raw",
    "image_name": "my_image",
    "force": "True"
  }
}
```

Response

Example. Create volume: JSON response

```json
{
  "os-volume_upload_image": {
    "status": "uploading",
    "image_id": "3dc6c4d9-95e9-4cdb-a076-cc3ed50b9654",
    "image_name": "20150430_volume_to_image-003",
    "volume_type": {
      "name": "M1",
      "qos_specs_id": null,
      "deleted": false,
      "created_at": "2014-10-11T11:26:56.000000",
      "updated_at": null,
      "deleted_at": null,
      "id": "13a0247c-9363-401d-acd3-c5d07ccd1a45"
    },
    "container_format": "bare",
    "size": 2,
    "disk_format": "raw",
    "id": "52102306-e352-4c84-9233-2ac2b511d29b",
    "display_description": null,
    "updated_at": "2015-04-30T06:31:31.000000"
  }
}
```

This table shows the body parameters for the create image response:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>status</td>
<td>String</td>
<td>Status of the volume. The status will become &quot;uploading&quot; immediately after image creation is started from the volume. When image creation is complete, the status will return to &quot;available&quot; or &quot;in-use&quot;.</td>
</tr>
<tr>
<td>image_id</td>
<td>uuid</td>
<td>ID of the created image.</td>
</tr>
<tr>
<td>image_name</td>
<td>String</td>
<td>Value specified in the Request Body.</td>
</tr>
<tr>
<td>volume_type</td>
<td>String</td>
<td>Type of the volume. When the volume type is not specified, &quot;null&quot; is returned instead of a child element.</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the volume type.</td>
</tr>
<tr>
<td>qos_specs_id</td>
<td>uuid</td>
<td>ID of the qos spec.</td>
</tr>
<tr>
<td>deleted</td>
<td>Boolean</td>
<td>Flag indicating that a volume has been deleted.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>---------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>created_at</td>
<td>Datetime</td>
<td>Creation datetime of the volume type.</td>
</tr>
<tr>
<td>updated_at</td>
<td>Datetime</td>
<td>Update datetime of the volume type.</td>
</tr>
<tr>
<td>deleted_at</td>
<td>Datetime</td>
<td>Deletion datetime of the volume type.</td>
</tr>
<tr>
<td>id</td>
<td>Uuid</td>
<td>ID of the volume type.</td>
</tr>
<tr>
<td>container_format</td>
<td>String</td>
<td>Value specified (or omitted) in the Request Body.</td>
</tr>
<tr>
<td>size</td>
<td>Int</td>
<td>Volume size, in GB.</td>
</tr>
<tr>
<td>disk_format</td>
<td>String</td>
<td>The value specified (or omitted) in the Request Body.</td>
</tr>
<tr>
<td>id</td>
<td>Uuid</td>
<td>ID of the volume.</td>
</tr>
<tr>
<td>display_description</td>
<td>String</td>
<td>Description of the volume.</td>
</tr>
<tr>
<td>updated_at</td>
<td>Datetime</td>
<td>Update datetime of the volume.</td>
</tr>
</tbody>
</table>

### 2.1.6.20 Extend volume

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/volumes/{volume_id}/action</td>
<td>Extend size of volume</td>
</tr>
</tbody>
</table>

Normal response codes: 202

⚠️ **Caution**

It is necessary to restart the operating system or create a file system for the new size to be recognized by the operating system.

To guarantee proper operation, use this API after shelving an existing space that is in use.

#### Request

This table shows the URI parameters for the extend size of volume request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
<tr>
<td>{volume_id}</td>
<td>UUID</td>
<td>The unique identifier of an existing volume.</td>
</tr>
</tbody>
</table>

This table shows the body parameters for the extend size of volume request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new_size</td>
<td>Integer</td>
<td>Specifies the size to extend the volume to.</td>
</tr>
</tbody>
</table>

#### Example. Create volume: JSON request

```json
{
  "os-extend": {
    "new_size": 20
  }
}
```
Response

This operation does not return a response body.
2.2 Extended storage

2.2.1 Extended storage

The APIs below can be used.

- APIs used in 『API Reference Manual（Application Platform Service）』 — 『Template/Development environment』 — 『Orchestration』.
- "Volume attachment" category APIs listed in API list on page 7.
2.3 Backup

2.3.1 Restrictions

This feature is not supported as at the time of issue of this document.

When using backup products

The APIs listed in API list on page 235 can be used.
2.4 Snapshot

2.4.1 Generate URLs when using APIs

The APIs require URLs of the blockstoragev2 type, which can be generated by the identity service on the Service catalog.

The endpoint URL is returned in the following format by the identity service.

https://hostName/v2/{tenant_id}

Create the URL in one of the following formats:
- If you remember the tenant_id:
  Join the path name of each API in the host section of the end point URL
- If you do not remember the tenant_id:
  The endpoint URL and the path name of each API with "v2/{tenant_id}" removed from the beginning

2.4.2 API list

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | GET /v2/{tenant_id}/snapshots | List snapshots
      | GET /v2/{tenant_id}/snapshots/detail | List snapshots (detailed)

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
</table>
| 2    | GET /v2/{tenant_id}/snapshots | Retrieves a list of the summary information about snapshots of all accessible storage blocks
      | GET /v2/{tenant_id}/snapshots/detail | Retrieves a list of detailed information about snapshots of all accessible storage blocks

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
</table>
| 3    | GET /v2/{tenant_id}/snapshots/{snapshot_id} | Show snapshot information
      | DELETE /v2/{tenant_id}/snapshots/{snapshot_id} | Delete snapshot

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
</table>
| 4    | POST /v2/{tenant_id}/snapshots | Creates a point-in-time snapshot of a volume

Block Storage API v2 Snapshots
<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>PUT /v2/{tenant_id}/snapshots/{snapshot_id}</td>
<td>Update snapshot</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Updates the specified snapshot</td>
</tr>
</tbody>
</table>

### Block Storage API v2 Snapshot actions

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>POST /v2/{tenant_id}/snapshots/{snapshot_id}/action</td>
<td>Restore volume from the snapshot</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Restores a volume from a snapshot</td>
</tr>
</tbody>
</table>

#### 2.4.3 Request header

<table>
<thead>
<tr>
<th>Header</th>
<th>Specified value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accept</td>
<td>application/json</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>X-Auth-Token</td>
<td>Authentication token</td>
</tr>
</tbody>
</table>

#### 2.4.4 API error codes

<table>
<thead>
<tr>
<th>Error code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>500, other codes possible</td>
<td>Server Error</td>
</tr>
<tr>
<td>400</td>
<td>badRequest</td>
</tr>
<tr>
<td>401</td>
<td>unauthorized</td>
</tr>
<tr>
<td>403</td>
<td>Forbidden, resizeNotAllowed</td>
</tr>
<tr>
<td>404</td>
<td>itemNotFound</td>
</tr>
</tbody>
</table>

#### 2.4.5 Notes

When an API (snapshot list, etc.) used to display a list of resources is executed, only some of the availability zone information may be returned. If this happens, it is assumed that infrastructure maintenance is in progress, so wait for a few moments (at least one minute) and then execute the API again.
2.4.6 API details

2.4.6.1 List snapshots

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/snapshots</td>
<td>Lists summary information for all Block Storage snapshots that the tenant who submits the request can access.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the list snapshots request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. List snapshots: JSON response

```json
{
    "snapshots": [
        {
            "status": "available",
            "description": "Very important",
            "created_at": "2013-02-25T04:13:17.000000",
            "metadata": {},
            "volume_id": "5aa119a8-d25b-45a7-8d1b-88e127885635",
            "size": 1,
            "id": "2bb856e1-b3d8-4432-a858-09e4ce939389",
            "name": "snap-001"
        },
        {
            "status": "available",
            "description": "Weekly backup",
            "created_at": "2013-02-25T07:20:38.000000",
            "metadata": {},
            "volume_id": "806092e3-7551-4fff-a005-49016f4943b1",
            "size": 1,
            "id": "e820db06-58b5-439d-bac6-c01faa3f6499",
            "name": "snap-002"
        }
    ]
}
```
2.4.6.2 List snapshots (detailed)

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/snapshots/detail</td>
<td>Lists detailed information for all Block Storage snapshots that the tenant who submits the request can access.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the list snapshots (detailed) request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

**Response**

**Example. List snapshots (detailed): JSON response**

```json
{
    "snapshots": [
        {
            "status": "available",
            "os-extended-snapshot-attributes:progress": "100%",
            "description": "Daily backup",
            "created_at": "2013-02-25T07:30:12.000000",
            "metadata": {},
            "volume_id": "5aa119a8-d25b-45a7-8d1b-88e127885635",
            "os-extended-snapshot-attributes:project_id": "0c2eba2c5af04d3f9e9d0d410b371fde",
            "size": 30,
            "id": "43f20e0e-2c2c-4770-9d4e-c3d769ae5470",
            "name": "snap-001",
            "availability_zone":"jp-east-1a"
        },
        {
            "status": "available",
            "os-extended-snapshot-attributes:progress": "100%",
            "description": "Weekly backup",
            "created_at": "2013-02-25T07:20:38.000000",
            "metadata": {},
            "volume_id": "806092e3-7551-4fff-a005-49016f4943b1",
            "os-extended-snapshot-attributes:project_id": "0c2eba2c5af04d3f9e9d0d410b371fde",
            "size": 1,
            "id": "e820db06-58b5-439d-bac6-c01faa3f6499",
            "name": "snap-002",
            "availability_zone":"jp-east-1a"
        }
    ]
}
```
2.4.6.3 Show snapshot information

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/v2/{tenant_id}/snapshots/ {snapshot_id}</td>
<td>Shows information for a specified snapshot.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

Request

This table shows the URI parameters for the show snapshot information request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
<tr>
<td>{snapshot_id}</td>
<td>UUID</td>
<td>The unique identifier of an existing snapshot.</td>
</tr>
</tbody>
</table>

This operation does not require a request body.

Response

Example. Show snapshot information: JSON response

```json
{
    "snapshot": {
        "status": "available",
        "os-extended-snapshot-attributes:progress": "100%",
        "description": "Daily backup",
        "created_at": "2013-02-25T04:13:17.000000",
        "metadata": {}
    },
    "volume_id": "5aa119a8-d25b-45a7-8d1b-88e127885635",
    "os-extended-snapshot-attributes:project_id": "0c2eba2c5af04d3f9e9d0410b371fde",
    "size": 1,
    "id": "2bb85611-b3d8-4432-a858-09e4ce939389",
    "name": "snap-001",
    "availability_zone": "jp-east-1a"
}
```

2.4.6.4 Delete snapshot

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DELETE</td>
<td>/v2/{tenant_id}/snapshots/ {snapshot_id}</td>
<td>Deletes a specified snapshot.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

Request

This table shows the URI parameters for the delete snapshot request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
</tbody>
</table>
This operation does not accept a request body and does not return a response body.

### 2.4.6.5 Create snapshot

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td><code>/v2/{tenant_id}/snapshots</code></td>
<td>Creates a snapshot, which is a point-in-time copy of a volume. You can create a volume from the snapshot.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

#### Request

This table shows the URI parameters for the create snapshot request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>{tenant_id}</code></td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
</tbody>
</table>

This table shows the query parameters for the create snapshot request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>volume_id</code></td>
<td>String</td>
<td>To create a snapshot from an existing volume, specify the ID of the existing volume.</td>
</tr>
<tr>
<td><code>force</code></td>
<td>Boolean</td>
<td>[True/False] Indicate whether to snapshot, even if the volume is attached. Default==False.</td>
</tr>
<tr>
<td><code>name</code></td>
<td>String</td>
<td>Name of the snapshot. Default==None.</td>
</tr>
<tr>
<td><code>description</code></td>
<td>String</td>
<td>Description of snapshot. Default==None.</td>
</tr>
</tbody>
</table>

#### Example. Create snapshot: JSON request

```json
{
  "snapshot": {
    "name": "snap-001",
    "description": "Daily backup",
    "volume_id": "5aa119a8-d25b-45a7-8d1b-88e127885635",
    "force": true
  }
}
```
Response

**Example. Create snapshot: JSON response**

```json
{
    "snapshot": {
        "status": "creating",
        "description": "Daily backup",
        "created_at": "2013-02-25T03:56:53.081642",
        "metadata": {},
        "volume_id": "5aa119a8-d25b-45a7-8d1b-88e127885635",
        "size": 1,
        "id": "ffa9bc5e-1172-4021-acaf-cdcd78a9584d",
        "name": "snap-001"
    }
}
```

### 2.4.6.6 Update snapshot

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUT</td>
<td>/v2/{tenant_id}/snapshots/ {snapshot_id}</td>
<td>Updates a specified snapshot.</td>
</tr>
</tbody>
</table>

Normal response codes: 200

**Request**

This table shows the URI parameters for the update snapshot request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
<tr>
<td>{snapshot_id}</td>
<td>UUID</td>
<td>The unique identifier of an existing snapshot.</td>
</tr>
</tbody>
</table>

This table shows the query parameters for the update snapshot request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String (Optional)</td>
<td>Describes the snapshot.</td>
</tr>
<tr>
<td>name</td>
<td>String (Optional)</td>
<td>The name of the snapshot.</td>
</tr>
</tbody>
</table>

**Example. Update snapshot: JSON request**

```json
{
    "snapshot": {
        "name": "snap-002",
        "description": "This is yet, another snapshot."
    }
}
```
2.4.6.7 Restore volume from the snapshot

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/v2/{tenant_id}/snapshots/{snapshot_id}/action</td>
<td>Restore a volume from the snapshot.</td>
</tr>
</tbody>
</table>

Normal response codes: 202

Request

This table shows the URI parameters for the restore volume from the snapshot request:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{tenant_id}</td>
<td>String</td>
<td>The unique identifier of the tenant or account.</td>
</tr>
<tr>
<td>{snapshot_id}</td>
<td>UUID</td>
<td>The unique identifier of an existing snapshot.</td>
</tr>
</tbody>
</table>

Example. Restore volume from the snapshot: JSON request

```json
{
  "fcx-restore": {}
}
```

Response

This operation does not return a response body.
2.5 Images

2.5.1 Images

The APIs below can be used.

- "Images" category APIs listed in API list on page 7.
- "Metadata" category APIs listed in API list on page 7.
- "Images with size attribute" category APIs listed in API list on page 181.
- "Image Service API v2 Images" category APIs listed in API list on page 181.
- "Image Service API v2 Image data" category APIs listed in API list on page 181.
2.6 Extended storage replication

2.6.1 Extended storage replication

The APIs below can be used.

- APIs listed in *Extended storage* on page 233
- APIs listed in *API list* on page 235
## 2.7 Object storage

### 2.7.1 Generate URLs when using APIs

The APIs require URLs of the object-store type, which can be generated by the identity service on the Service catalog.

The endpoint URL is returned in the following format by the identity service.

```
https://hostName/v1/AUTH_{project_id}
```

In the descriptions that follow, "AUTH_{project_id}" is referred to as "{account}".

### 2.7.2 API list

<table>
<thead>
<tr>
<th>Item</th>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GET /v1/{account}{?limit,marker,end_marker,format,prefix,delimiter} Retrieve container list</td>
<td>Retrieves the account details and a list of the containers.</td>
</tr>
<tr>
<td>2</td>
<td>POST /v1/{account} Update account metadata</td>
<td>Creates, updates, or deletes the account metadata.</td>
</tr>
<tr>
<td>3</td>
<td>HEAD /v1/{account} Retrieve account metadata</td>
<td>Retrieves the account metadata.</td>
</tr>
<tr>
<td>4</td>
<td>GET /v1/{account}/{container}{?limit,marker,end_marker,prefix,format,delimiter,path} Retrieve a list of objects</td>
<td>Retrieves the container details and a list of objects.</td>
</tr>
<tr>
<td>5</td>
<td>PUT /v1/{account}/{container} Create container</td>
<td>Creates a container.</td>
</tr>
<tr>
<td>6</td>
<td>DELETE /v1/{account}/{container} Delete container</td>
<td>Deletes an empty container.</td>
</tr>
<tr>
<td>7</td>
<td>POST /v1/{account}/{container} Update container metadata</td>
<td>Deletes, creates, or updates the custom metadata of a container.</td>
</tr>
<tr>
<td>8</td>
<td>HEAD /v1/{account}/{container} Retrieve container metadata</td>
<td>Retrieves the container metadata, including the number of objects and their size in bytes</td>
</tr>
<tr>
<td>9</td>
<td>GET /v1/{account}/{container}/{object}{?multipart-manifest,signature,expires} Retrieve object</td>
<td>Retrieves object contents and metadata.</td>
</tr>
<tr>
<td>Item</td>
<td>API</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>-------------</td>
</tr>
<tr>
<td>10</td>
<td>PUT /v1/{account}/{container}/{object}{?multipart-manifest,signature,expires} Create object</td>
<td>Creates an object using the specified content and metadata or replaces an existing object with the specified content and metadata.</td>
</tr>
<tr>
<td>11</td>
<td>COPY /v1/{account}/{container}/{object} Copy object</td>
<td>Copies the specified object.</td>
</tr>
<tr>
<td>12</td>
<td>DELETE /v1/{account}/{container}/{object}{?multipart-manifest} Delete object</td>
<td>Deletes an object.</td>
</tr>
<tr>
<td>13</td>
<td>HEAD /v1/{account}/{container}/{object}{?signature,expires} Retrieve object metadata</td>
<td>Retrieves object metadata.</td>
</tr>
<tr>
<td>14</td>
<td>POST /v1/{account}/{container}/{object} Update object metadata</td>
<td>Creates or updates object metadata.</td>
</tr>
</tbody>
</table>

### 2.7.3 General requirements

If a value in the request parameter contains a character that cannot be used as is in the URL, it must be encoded using UTF-8.

### 2.7.4 API details

#### 2.7.4.1 List containers

**GET /v1/{account}{?limit,marker,end_marker,format,prefix,delimiter}**

Retrieves the account details and a list of its containers.

If a format is not specified, the container list will be returned in text/plain format.

Also, when the query parameter is used, the lists in a container can be retrieved, divided by page. When the number of containers retrieved is smaller than the value specified in the limit parameter, a list right to end will be retrieved. When the number of containers retrieved equals the value specified in the limit parameter, containers that are yet to be retrieved remain in the list.

When the list of containers is retrieved successfully, the following status code is returned:

- 200 OK: The container list is included in the response body.
- 204 No Content: The container does not exist or the result of filtering using the limit, marker or end_marker query parameters is empty.

**Request headers**

- X-Auth-Token

Authentication token
<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Newest**

When "True" is set, object storage searches for and returns the newest replica out of all of the replicas. If this header is omitted, object storage selects one normal replica and returns it. When "True" is specified in this header, the response time becomes longer. Only use this item when absolutely necessary.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Accept**

Valid values: application/json, application/xml, and text/xml.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..*</td>
</tr>
</tbody>
</table>

**Request parameters**

{account}

Name uniquely assigned by project

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**limit**

Maximum number of containers to retrieve at once.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**marker**

Returns a list of container names that follow the specified string.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**end_marker**

Returns a list of container names that precede the specified string.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**format**

Specifies the response format. Valid values: json, xml, and plain. The default is plain. When format=xml or format=json is specified, the response will include not only the container name but also other details.
When format=plain is specified, a list of container names delimited by line feeds will be returned.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**prefix**

Returns a list of containers that start with this string.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

### Response headers

**Content-Length**

Number of bytes of the response body

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Content-Type**

MIME type of the response body

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Account-Object-Count**

Number of objects in the account

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Account-Bytes-Used**

Size (in bytes) of the objects stored in object storage by the account

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Account-Container-Count**

Number of containers

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Account-Meta-name**

Account metadata. \{name\} is the name of the metadata item.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..*</td>
</tr>
</tbody>
</table>
X-Account-Meta-Temp-URL-Key
Secret key used for temporary URLs. If not set, this header will not be returned.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

X-Account-Meta-Temp-URL-Key-2
Second secret key used for temporary URLs. If not set, this header will not be returned.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

X-Trans-Id
ID assigned to this request. This is used when inquiring about issues.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uuid</td>
<td>1..1</td>
</tr>
</tbody>
</table>

Date
Datetime when transaction was executed

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datetime</td>
<td>1..1</td>
</tr>
</tbody>
</table>

Response elements

account
Envelope of the response

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
<th>Parent element</th>
<th>Child element</th>
</tr>
</thead>
<tbody>
<tr>
<td>account</td>
<td>1..1</td>
<td>None</td>
<td>container</td>
</tr>
</tbody>
</table>

container
Set of container information

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
<th>Parent element</th>
<th>Child element</th>
</tr>
</thead>
<tbody>
<tr>
<td>container</td>
<td>0..n</td>
<td>account</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>count</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>bytes</td>
</tr>
</tbody>
</table>

ame
Container name

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
<th>Parent element</th>
<th>Child element</th>
</tr>
</thead>
<tbody>
<tr>
<td>string</td>
<td>0..n</td>
<td>container</td>
<td>None</td>
</tr>
</tbody>
</table>

count
Number of objects held by the container

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
<th>Parent element</th>
<th>Child element</th>
</tr>
</thead>
<tbody>
<tr>
<td>string</td>
<td>0..1</td>
<td>container</td>
<td>None</td>
</tr>
</tbody>
</table>

**bytes**

Size of the objects held by the container

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
<th>Parent element</th>
<th>Child element</th>
</tr>
</thead>
<tbody>
<tr>
<td>string</td>
<td>0..1</td>
<td>container</td>
<td>None</td>
</tr>
</tbody>
</table>

**Example List containers response: HTTP and JSON**

**Example of request**

```
curl -i $publicURL?format=json -X GET -H "X-Auth-Token: $token"
```

**Example of response**

```
HTTP/1.1 200 OK
Content-Length: 96
X-Account-Object-Count: 1
X-Timestamp: 1389453423.35964
X-Account-Meta-Subject: Literature
X-Account-Bytes-Used: 14
X-Account-Container-Count: 2
Content-Type: application/json; charset=utf-8
Accept-Ranges: bytes
X-Trans-Id: tx274a77a8975c4a66aeb24-0052d95365
Date: Fri, 17 Jan 2014 15:59:33 GMT

[{
   "count": 0,
   "bytes": 0,
   "name": "janeausten"
},
{
   "count": 1,
   "bytes": 14,
   "name": "marktwain"
}]
```

**Example List containers response: HTTP and XML**

**Example of request**

```
curl -i $publicURL?format=xml -X GET -H "X-Auth-Token: $token"
```

**Example of response**

```
HTTP/1.1 200 OK
```
2.7.4.2 Update account metadata POST /v1/{account}

Creates, updates or deletes account metadata items

Specify the X-Account-Meta-{name} header to create, update, or delete a metadata item, where {name} is its name.

If a specified {name} matches the name of an existing metadata item, then it will be overwritten.

To delete a metadata item, either issue a request with an empty value in the header or specify the "X-Remove-Account-Meta-{name}: anyValue" header (for example, "X-Remove-Account-Meta-Book: x" - in this case, the arbitrary value section will be ignored).

Existing metadata items not specified using this API will remain unchanged.

The request body will not be accepted.

Upon successful completion, the 204 status code will be returned.

After updating, perform an account metadata retrieval request to check if the changes took effect.

Request headers

X-Auth-Token
Authentication token

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

X-Account-Meta-Temp-URL-Key
Secret key used for temporary URLs.
X-Account-Meta-Temp-URL-Key-2
Second secret key used for temporary URLs. By using two keys, key rotation is possible.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

X-Account-Meta-name
Account metadata. In {name}, specify the name of the metadata item to be created, updated, or deleted. To delete an item, leave the value empty in the header.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..*</td>
</tr>
</tbody>
</table>

Request parameters
(account)
Name uniquely assigned by project

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

Response headers

Content-Length
If the request was successful, this will be 0. If the request failed, this will be the size (in bytes) of the error text returned in the response body.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

Content-Type
If the request failed, this will be the MIME type of the error text returned in the response body.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

X-Trans-Id
ID assigned to this request. This is used when inquiring about issues.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uuid</td>
<td>1..1</td>
</tr>
</tbody>
</table>

Date
Datetime of request execution.
<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datetime</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Response elements**

None.

**Create account metadata items**

**Example of request**

```bash
```

**Example of response**

```
HTTP/1.1 204 No Content
Content-Length: 0
Content-Type: text/html; charset=UTF-8
X-Trans-Id: tx8c2dd6ae35442a4a5646-0052d954fb
Date: Fri, 17 Jan 2014 16:06:19 GMT
```

**Update account metadata items**

**Example of request**

```bash
```

**Example of response**

```
HTTP/1.1 204 No Content
Content-Length: 0
Content-Type: text/html; charset=UTF-8
X-Trans-Id: tx1439b96137364ab581156-0052d95532
Date: Fri, 17 Jan 2014 16:07:14 GMT
```

**Delete account metadata items**

**Example of request**

```bash
curl -i $publicURL -X POST -H "X-Auth-Token: $token" -H "X-Remove-Account-Meta-Subject: x"
```

**Example of response**

```
HTTP/1.1 204 No Content
Content-Length: 0
```
2.7.4.3 Retrieve account metadata HEAD /v1/{account}

Retrieves the account metadata

The account metadata comprises the following:

- Number of containers
- Number of objects
- Bytes used by object storage
- Custom metadata specified by the user

When processing the number of bytes used by object storage, handle with care. A large number of objects can be stored in the object storage. If possible, it is recommended to use a 64-bit unsigned integer.

The request body will not be accepted.

The metadata header must not be included in the request.

Upon successful completion, the 204 status code will be returned.

If the authentication token is not suitable, the 401 status code will be returned.

Request headers

**X-Auth-Token**

Authentication token

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Newest**

When "True" is set, object storage searches for and returns the newest replica out of all of the replicas. If this header is omitted, object storage selects one normal replica and returns it. When "True" is specified in this header, it becomes extremely costly. Only use this item when absolutely necessary.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boolean</td>
<td>0..1</td>
</tr>
</tbody>
</table>

Request parameter

**{account}**

Name that is uniquely assigned by project

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>
Response headers

**X-Account-Object-Count**
Number of objects in the account

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Account-Container-Count**
Number of containers

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Account-Bytes-Used**
Size of the objects stored in object storage by the account.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Account-Meta-name**
Account metadata. \{name\} is the name of the metadata item.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..*</td>
</tr>
</tbody>
</table>

**X-Account-Meta-Temp-URL-Key**
Secret key used for temporary URLs. If not set, this header will not be returned.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Account-Meta-Temp-URL-Key-2**
Second secret key used for temporary URLs. If not set, this header will not be returned.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Content-Length**
Number of bytes of the response body

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Content-Type**
MIME type of the response body

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>
X-Trans-Id
ID assigned to this request. This is used when inquiring about issues.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uuid</td>
<td>1..1</td>
</tr>
</tbody>
</table>

Date
Datetime when transaction was executed

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datetime</td>
<td>1..1</td>
</tr>
</tbody>
</table>

Response elements
None.

Example of request

curl -i $publicURL -X HEAD -H "X-Auth-Token: $token"

Example of response

HTTP/1.1 204 No Content
Content-Length: 0
X-Account-Object-Count: 1
X-Account-Meta-Book: MobyDick
X-Timestamp: 1389453423.35964
X-Account-Bytes-Used: 14
X-Account-Container-Count: 2
Content-Type: text/plain; charset=utf-8
Accept-Ranges: bytes
X-Trans-Id: txafb3504870144b8ca40f7-0052d955d4
Date: Fri, 17 Jan 2014 16:09:56 GMT

2.7.4.4 List objects GET /v1/{account}/{container}{?limit,marker,end_marker,prefix,format,delimiter,path}

Retrieves the container details and the list of objects

Object names can be specified in the query parameter.
If omitted, up to 10,000 object names stored in the container can be retrieved.

Also, when the query parameter is used, the lists in a container can be retrieved, divided by page. When the number of containers retrieved is smaller than the value specified in the limit parameter, a list right to end will be retrieved. When the number of containers retrieved equals the value specified in the limit parameter, containers that are yet to be retrieved remain in the list.

Upon successful completion, one of the following status code will be returned:
• 200 OK: If objects exist, a list of objects is returned.
• 204 No Content: The container does not exist or the result of filtering using the limit, marker or end_marker query parameters is empty.

If the specified container does not exist, 404 Not Found will be returned.

**Request headers**

**X-Auth-Token**

Authentication token

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Newest**

When "True" is set, object storage searches for and returns the newest replica out of all of the replicas. If this header is omitted, object storage selects one normal replica and returns it. When "True" is specified in this header, it becomes extremely costly. Only use this item when absolutely necessary.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Accept**

Valid values: application/json, application/xml, text/xml. This header is prioritized over the format query parameter.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Request parameters**

**{account}**

Name that is uniquely assigned by project

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**{container}**

Container name

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**limit**

Maximum number of lists to retrieve.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**marker**
Returns a list of object names that follow the specified string.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**end_marker**

Returns a list of object names that precede the specified string.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**prefix**

Prefix. Returns a list of objects that start with this string.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**format**

Specifies the response format. Valid values: json, xml, and plain. The default is plain.

When format=xml or format=json is specified, the response will include not only the object name but also other details.

When format=plain is specified, a list of object names delimited by line feeds will be returned.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**delimiter**

Delimiter for nested display of object names

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Char</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**path**

Returns a list of object names nested using a pseudo path. This is the same specifying “/” for "delimiter" and "{path}" for "prefix".

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Response headers**

**Content-Length**

Number of bytes of the response body

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Container-Object-Count**
Number of objects in the container

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Accept-Ranges**
Indicates that objects included in the body can be retrieved using the Range header.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Container-Meta-name**
Container metadata. {name} is the name of the metadata item.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Container-Bytes-Used**
Size (in bytes) of the objects in the container.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Content-Type**
MIME type of the response body.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Trans-Id**
ID assigned to this request. This is used when inquiring about issues.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uuid</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Date**
Datetime when transaction was executed.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datetime</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Response elements**
container
Envelope of the response

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
<th>Parent element</th>
<th>Child element</th>
</tr>
</thead>
<tbody>
<tr>
<td>container</td>
<td>1..1</td>
<td>None</td>
<td>object</td>
</tr>
</tbody>
</table>
**object**
Set of object information

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
<th>Parent element</th>
<th>Child element</th>
</tr>
</thead>
<tbody>
<tr>
<td>object</td>
<td>0..n</td>
<td>container</td>
<td></td>
</tr>
</tbody>
</table>

**name**
Object name

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
<th>Parent element</th>
<th>Child element</th>
</tr>
</thead>
<tbody>
<tr>
<td>string</td>
<td>1..1</td>
<td>object</td>
<td>None</td>
</tr>
</tbody>
</table>

**hash**
MD5 checksum of the object

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
<th>Parent element</th>
<th>Child element</th>
</tr>
</thead>
<tbody>
<tr>
<td>string</td>
<td>1..1</td>
<td>object</td>
<td>None</td>
</tr>
</tbody>
</table>

**bytes**
Size of the object

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
<th>Parent element</th>
<th>Child element</th>
</tr>
</thead>
<tbody>
<tr>
<td>string</td>
<td>1..1</td>
<td>object</td>
<td>None</td>
</tr>
</tbody>
</table>

**content_type**
MIME type of an object

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
<th>Parent element</th>
<th>Child element</th>
</tr>
</thead>
<tbody>
<tr>
<td>string</td>
<td>1..1</td>
<td>object</td>
<td>None</td>
</tr>
</tbody>
</table>

**last_modified**
Datetime when an object was created, or the datetime when the metadata was modified

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
<th>Parent element</th>
<th>Child element</th>
</tr>
</thead>
<tbody>
<tr>
<td>string</td>
<td>1..1</td>
<td>object</td>
<td>None</td>
</tr>
</tbody>
</table>

**Example Show container details response: HTTP and JSON**

**Example of request**
curl -i $publicURL/marktwain?format=json -X GET -H "X-Auth-Token:$token"
Example of response

HTTP/1.1 200 OK
Content-Length: 341
X-Container-Object-Count: 2
Accept-Ranges: bytes
X-Container-Meta-Book: TomSawyer
X-Timestamp: 1389727543.65372
X-Container-Bytes-Used: 26
Content-Type: application/json; charset=utf-8
X-Trans-Id: tx26377fe5fab74869825d1-0052d6bdf
Date: Wed, 15 Jan 2014 16:57:35 GMT

[
  {
    "hash": "451e372e48e0f6b1114fa0724aa79fa1",
    "last_modified": "2014-01-15T16:41:49.390270",
    "bytes": 14,
    "name": "goodbye",
    "content_type": "application/octet-stream"
  },
  {
    "hash": "ed076287532e863655e841e92bfc50d8c",
    "last_modified": "2014-01-15T16:37:43.427570",
    "bytes": 12,
    "name": "helloworld",
    "content_type": "application/octet-stream"
  }
]

Example Show container details response: HTTP and XML

Example of request

curl -i $publicURL/marktwain?format=xml -X GET -H "X-Auth-Token:$token"

Example of response

HTTP/1.1 200 OK
Content-Length: 500
X-Container-Object-Count: 2
Accept-Ranges: bytes
X-Container-Meta-Book: TomSawyer
X-Timestamp: 1389727543.65372
X-Container-Bytes-Used: 26
Content-Type: application/xml; charset=utf-8
X-Trans-Id: txc75ea9a6e6647d79e0c5-0052d6be76
Date: Wed, 15 Jan 2014 16:59:35 GMT

<?xml version="1.0" encoding="UTF-8"?>
<container name="marktwain">
  <object>
    <name>goodbye</name>
    <hash>451e372e48e0f6b1114fa0724aa79fa1</hash>
    <bytes>14</bytes>
    <content_type>application/octet-stream</content_type>
    <last_modified>2014-01-15T16:41:49.390270</last_modified>
  </object>
  <object>
    <name>helloworld</name>
    <hash>ed076287532e863655e841e92bfc50d8c</hash>
  </object>
</container>
### 2.7.4.5 Create container PUT /v1/{account}/{container}

Creates a container

It is not necessary to check if a container with the same name already exists before executing this API - if the container does not exist yet, it will be created, otherwise it will be updated.

If a container is created successfully, the 201 status code will be returned. If an existing container is updated successfully, the 202 status code will be returned.

#### Request headers

**X-Auth-Token**

Authentication token

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Read**

Sets the read permissions ACL.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..*</td>
</tr>
</tbody>
</table>

**X-Container-Write**

Sets the write permissions ACL.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..*</td>
</tr>
</tbody>
</table>

**X-Versions-Location**

Enables versioning of objects in a container. The name of another container that has been URL-encoded in UTF-8 format must be specified. To disable versioning, leave the value empty.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Meta-name**

Container metadata. \( \{ \text{name} \} \) is the name of the metadata item.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..*</td>
</tr>
</tbody>
</table>

**X-Container-Meta-Web-Index**
Sets the object name of the Index file. For example, when index.html is set, the index page will become /{container}/index.html.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Meta-Web-Error**

Sets the object name of error page files. For example, if error.html is set, 401 errors will become /{container}/401error.html, and 404 errors will become /{container}/404error.html objects.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Meta-Web-Listing**

When "TRUE" is set, directory list display becomes enabled. For security purposes, this item should not normally be enabled.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boolean</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Meta-Web-Listing-CSS**

Sets an object name of the style sheet for displaying directory lists. For example, if lists.css is set, the /{container}/lists.css object will be used as the style sheet.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Request parameters**

**{account}**

Name that is uniquely assigned by project

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**{container}**

Container name

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Response headers**

**Content-Length**

Number of bytes of the response body

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>
**Content-Type**
MIME type of the response body

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Trans-Id**
ID assigned to this request. This is used when inquiring about issues.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uuid</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Date**
Datetime when transaction was executed

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datetime</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Response elements**
None.

**When metadata is not set**

**Example of Request**

```bash
curl -i $publicURL/steven -X PUT -H "Content-Length: 0" -H "X-Auth-Token: $token"
```

**Example of response**

HTTP/1.1 201 Created
Content-Length: 0
Content-Type: text/html; charset=UTF-8
X-Trans-Id: tx7f6b7fa09bc2443a94df0-0052d58b56
Date: Tue, 14 Jan 2014 19:09:10 GMT

**When metadata is set**

**Example of request**

```bash
curl -i $publicURL/steven -X PUT -H "Content-Length: 0" -H "X-Auth-Token: $token"
```

**Example of response**

HTTP/1.1 201 Created
Content-Length: 0
Content-Type: text/html; charset=UTF-8
2.7.4.6 Delete container DELETE /v1/{account}/{container}

Deletes an empty container.

This operation will fail if the container contains any objects.

Upon successful completion, the 204 No Content status code will be returned.
If the operation fails, the 404 No Content or 409 Conflict status code will be returned.

Request header

<table>
<thead>
<tr>
<th>X-Auth-Token</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authentication token</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

Request parameters

{account}
Name that is uniquely assigned by project

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

{container}
Container name

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

Response headers

Content-Length
Number of bytes of the response body

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

Content-Type
MIME type of the response body

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

X-Trans-Id
ID assigned to this request. This is used when inquiring about issues.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uuid</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Date**

Datetime when transaction was executed

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datetime</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Response elements**

None.

**Example of request**

curl -i $publicURL/steven -X DELETE -H "X-Auth-Token: $token"

**Example of response**

Response when the container does not exist

HTTP/1.1 404 Not Found
Content-Length: 70
Content-Type: text/html; charset=UTF-8
X-Trans-Id: tx4d728126b17b43b598bf7-0052d81e34
Date: Thu, 16 Jan 2014 18:00:20 GMT

Response when the container existed and deletion was successful

HTTP/1.1 204 No Content
Content-Length: 0
Content-Type: text/html; charset=UTF-8
X-Trans-Id: txf76c375ebece4df19c84c-0052d81f14
Date: Thu, 16 Jan 2014 18:04:04 GMT

Response when the container exists but is not empty

HTTP/1.1 409 Conflict
Content-Length: 95
Content-Type: text/html; charset=UTF-8
X-Trans-Id: tx7782dc6a97b94a46956b5-0052d81f6b
Date: Thu, 16 Jan 2014 18:05:31 GMT
<html><h1>Conflict</h1><p>There was a conflict when trying to complete your request.</p></html>

2.7.4.7 Update container metadata POST /v1/{account}/ {container}

Creates, updates or deletes custom container metadata items
Specify the X-Container-Meta-{name} header to create, update, or delete a metadata item. where {name} is its name.

If a specified {name} matches the name of an existing metadata item, then it will be overwritten.

To delete a metadata item, either issue a request with an empty value in the header or specify the "X-Remove-Container-Meta-{name}: anyvalue" header. (for example, "X-Remove-Container-Meta-Book: x" - in this case, the arbitrary value section will be ignored).

Existing metadata items not specified using this API will remain unchanged.

Upon successful completion, the 204 status code will be returned.

After updating, perform a container metadata retrieval request to check if the changes took effect.

**Request headers**

**X-Auth-Token**
Authentication token

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Read**
Sets the read permissions ACL.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Remove-Container-name**
**X-Remove-Container-Meta-name**
Deletes the {name} metadata item.

For example, if "X-Remove-Container-Read" is specified, the "X-Container-Read" metadata item will be deleted.

For custom metadata, if "X-Remove-Container-Meta-Book" is specified, the "X-Remove-Container-Meta-Book" metadata item will be deleted.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Write**
Sets the write permissions ACL.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Versions-Location**
Enables versioning of objects in a container. The name of another container that has been URL-encoded in UTF-8 format must be specified. To disable versioning, leave the value empty.
<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Remove-Versions-Location**
Disables versioning.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Meta-name**
Container metadata. (name) is the name of the metadata item.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..*</td>
</tr>
</tbody>
</table>

**X-Container-Meta-Web-Index**
Sets the object name of the Index file. For example, when index.html is set, the index page will become /{container}/index.html.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Meta-Web-Error**
Sets the object name of error page files. For example, if error.html is set, 401 errors will become /{container}/401error.html, and 404 errors will become /{container}/404error.html.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Meta-Web-Listing**
When "TRUE" is set, directory list display becomes enabled. For security purposes, this item should not normally be enabled.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boolean</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Meta-Web-Listing-CSS**
Sets an object name of the style sheet for displaying directory lists. For example, if lists.css is set, the /{container}/lists.css object will be used as the style sheet.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Request parameters**

{account}
Name uniquely assigned by project
<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

{container}
Container name

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Response headers**

**Content-Length**
Number of bytes of the response body

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Content-Type**
MIME type of the response body

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Trans-Id**
ID assigned to this request. This is used when inquiring about issues.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uuid</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Date**
Datetime when transaction was executed

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datetime</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Response elements**
None.

**Create container metadata:**

**Example of request**

```
```
Example of response

HTTP/1.1 204 No Content
Content-Length: 0
Content-Type: text/html; charset=UTF-8
X-Trans-Id: tx05dbd434c651429193139-0052d82635
Date: Thu, 16 Jan 2014 18:34:29 GMT

Update container metadata:

Example of request

curl -i $publicURL/marktwain -X POST -H "X-Auth-Token: $token" -H "X-Container-Meta-Author: SamuelClemens"

Example of response

HTTP/1.1 204 No Content
Content-Length: 0
Content-Type: text/html; charset=UTF-8
X-Trans-Id: txe60c7314bf614bb39dfe4-0052d82653
Date: Thu, 16 Jan 2014 18:34:59 GMT

Delete container metadata:

Example of request


Example of response

HTTP/1.1 204 No Content
Content-Length: 0
Content-Type: text/html; charset=UTF-8
X-Trans-Id: tx7997e18da2a34a9e84ceb-0052d826d0
Date: Thu, 16 Jan 2014 18:37:04 GMT

2.7.4.8 Retrieve container metadata HEAD /v1/{account}/{container}

Retrieves the container metadata, including the number of objects and their size in bytes.
Upon successful completion, the 204 status code is returned.

Request headers
  X-Auth-Token
  Authentication token
### Data type

<table>
<thead>
<tr>
<th>Field</th>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td></td>
<td>0..1</td>
</tr>
</tbody>
</table>

### X-Newest

When "True" is set, object storage searches for and returns the newest replica out of all of the replicas. If this header is omitted, object storage selects one normal replica and returns it. When "True" is specified in this header, it becomes extremely costly. Only use this item when absolutely necessary.

<table>
<thead>
<tr>
<th>Field</th>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td></td>
<td>0..1</td>
</tr>
</tbody>
</table>

### Request parameters

#### {account}

Name uniquely assigned by project

<table>
<thead>
<tr>
<th>Field</th>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td></td>
<td>1..1</td>
</tr>
</tbody>
</table>

#### {container}

Container name

<table>
<thead>
<tr>
<th>Field</th>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td></td>
<td>1..1</td>
</tr>
</tbody>
</table>

### Response headers

#### Content-Length

Number of bytes of the response body

<table>
<thead>
<tr>
<th>Field</th>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td></td>
<td>1..1</td>
</tr>
</tbody>
</table>

#### X-Container-Object-Count

Number of objects in the container

<table>
<thead>
<tr>
<th>Field</th>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td></td>
<td>1..1</td>
</tr>
</tbody>
</table>

#### X-Container-Meta-name

Container metadata. (name) is the name of the metadata item.

<table>
<thead>
<tr>
<th>Field</th>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td></td>
<td>1..1</td>
</tr>
</tbody>
</table>

#### X-Container-Bytes-Used

Size (in bytes) of the objects in the container
<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Read**
Read permissions ACL. If omitted, this will not be returned.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Write**
Write permissions ACL. If omitted, this will not be returned.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Versions-Location**
Versioning settings of objects in a container. If omitted, this will not be returned.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Content-Type**
MIME type of the response body

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Trans-Id**
ID assigned to this request. This is used when inquiring about issues.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uuid</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Date**
Datetime when transaction was executed.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datetime</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Meta-Web-Index**
Object name of Index file.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Meta-Web-Error**
Object name of Error page file.
<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Meta-Web-Listing**

When display of directory lists is enabled, "True" is returned.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Container-Meta-Web-Listing-CSS**

Object name of the style sheet for displaying directory lists.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Response elements**

None.

**Example of request**

```
curl -i $publicURL/marktwain -X HEAD -H "X-Auth-Token: $token"
```

**Example of response**

```
HTTP/1.1 204 No Content
Content-Length: 0
Accept-Ranges: bytes
X-Container-Meta-Book: TomSawyer
X-Timestamp: 1389727543.65372
X-Container-Meta-Author: SamuelClemens
X-Container-Bytes-Used: 14
Content-Type: text/plain; charset=utf-8
X-Trans-Id: tx0287b982a268461b9ec14-0052d826e2
Date: Thu, 16 Jan 2014 18:37:22 GMT
```

**2.7.4.9 Download object GET /v1/{account}/{container}/{object}**

Retrieves object contents and their metadata.

When handling large objects, the response will include the object name after joining. When retrieving the manifest file of a static large object as is, use the multipart-manifest query parameter.

Upon successful completion, the 200 status code is returned. If the object does not exist, the 404 status code is returned.
## Request headers

### X-Auth-Token
Authentication token

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

### X-Newest
When "True" is set, object storage searches for and returns the newest replica out of all of the replicas. If this header is omitted, object storage selects one normal replica and returns it. When "True" is specified in this header, it becomes extremely costly. Only use this item when absolutely necessary.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

### Range
Range of contents to be retrieved.
You can use the Range header to specify a range to retrieve part of the data. When multiple ranges are specified, separate them with a comma.
The method for specifying a range is as follows:
- **Byte range:**
  Specify from which byte to which byte of the data is to be returned. When the end byte of the data is omitted, the data up to the end will be returned.
- **Suffix byte range:**
  Specify how many bytes to return for the data suffix.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dict</td>
<td>0..1</td>
</tr>
</tbody>
</table>

### If-Match
Refer to [http://www.ietf.org/rfc/rfc2616.txt](http://www.ietf.org/rfc/rfc2616.txt)

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dict</td>
<td>0..1</td>
</tr>
</tbody>
</table>

### If-Modified-Since
Refer to [http://www.ietf.org/rfc/rfc2616.txt](http://www.ietf.org/rfc/rfc2616.txt)

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dict</td>
<td>0..1</td>
</tr>
</tbody>
</table>

### If-Unmodified-Since
Refer to [http://www.ietf.org/rfc/rfc2616.txt](http://www.ietf.org/rfc/rfc2616.txt)

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dict</td>
<td>0..1</td>
</tr>
</tbody>
</table>
## Request parameters

**{account}**  
Name uniquely assigned by project

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**{container}**  
Container name

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**{object}**  
Object name

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Signature**  
Uses a signature for requests in the URL feature with effective period.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Expires**  
Specifies the effective period of the signature in the URL feature with effective period.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**multipart-manifest**  
When the object is large, multipart-manifest=get can be specified in the query parameter. The object itself is not returned at that time. Instead, in the case of a dynamic large object, the X-Object-Manifest response header is returned, and in the case of a static large object, Manifest is returned in the response body.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

## Response headers

**Content-Length**  
Size of object (bytes)

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Accept-Ranges**
Ranges type that can be specified when retrieving objects

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

Last-Modified
Datetme when an object was created, or the datetime when the metadata was last modified

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

ETag
When an object is smaller than 5 GB, this value will be the MD5 checksum of the object. Also, this value is not enclosed in quotations.
In the case of a manifest object, a value with the retrieved MD5 checksum for a string that combines the ETag and MD5 checksums of each segment of the manifest is returned.
You are strongly advised to compare the ETag header value with the MD5 checksum value of the object that was actually downloaded. If the values are different, this indicates that the contents are corrupted, so perform retry processing.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

Content-Type
MIME type of the response body

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

Content-Encoding
Metadata of Content-Encoding.
If omitted, this will not be returned.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

Content-Disposition
This is returned when the browser behavior has been set.
Refer to "http://www.ietf.org/rfc/rfc2183.txt" for details on the value that is returned.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

X-Delete-At
The date on which an object will be deleted is returned in UNIX Epoch timestamp format. If omitted, this will not be returned.
<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Object-Meta-name**

Object metadata {name} is the name of the metadata item.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Object-Manifest**

Returns the dynamic large object settings. This value takes the container name and prefix name of the split object, as follows. container/prefix

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Static-Large-Object**

If the object is the manifest object of a static large object, "True" will be returned.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bool</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Trans-Id**

ID assigned to this request. This is used when inquiring about issues.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uuid</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Date**

Datetime when transaction was executed

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datetime</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Response elements**

The object is returned.

**When an object exists**

**Example of request**

```bash
curl -i $publicURL/marktwain/goodbye -X GET -H "X-Auth-Token: $token"
```

**Example of response**

```
HTTP/1.1 200 OK
Content-Length: 14
```
When an object does not exist

Example of request

curl -i $publicURL/janeausten/goodbye -X GET -H "X-Auth-Token: $token"

Example of response

HTTP/1.1 404 Not Found
Content-Length: 70
Content-Type: text/html; charset=UTF-8
X-Trans-Id: tx073f7cbb850c4c9934b9-0052d82b04
Date: Thu, 16 Jan 2014 18:55:00 GMT
<html><h1>Not Found</h1><p>The resource could not be found.</p></html>

2.7.4.10 Upload object PUT /v1/{account}/{container}/{object} {?multipart-manifest,signature,expires}

Creates an object using the specified content and metadata or replaces an existing object with the specified content and metadata.

When this operation is used to copy a manifest object, the new object becomes a normal object with all segments joined. Therefore, it is not possible to copy an object that has a combined size exceeding 5 GB.

When creation of an object is successful, the 201 Created status code is returned.
When a request times out, the 408 Request Timeout error code is returned.
If neither Transfer-Encoding nor Content-Length request header are specified, the 411 Length Required error code is returned.
If the value set for ETag and the MD5 checksum of the data do not match, the 422 Unprocessable Entity error code is returned.

Request headers

**X-Object-Manifest**
Sets the manifest object of a dynamic large object. This value takes the container name and prefix name of the split object as follows: {container}/{prefix}. Also, it is necessary for the container name and prefix name of the split object to be UTF-8-encoded as well as URL-encoded before being set in the header.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>
X-Auth-Token
Authentication token

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

Content-Length
Size of object (bytes). This item cannot be set if transferring in chunked format.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

Transfer-Encoding
If transferring in chunked format, use Transfer-Encoding: chunked. If using this header, a Content-Length header cannot be assigned.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

Content-Type
Changes the MIME type of an object.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

X-Detect-Content-Type
When "true" is set, the object storage will automatically set the MIME type based on the file extension. Settings using the Content-Type header will be ignored.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boolean</td>
<td>0..1</td>
</tr>
</tbody>
</table>

X-Copy-From
Specifies the copy source object name.
The value is specified in {container}/{object} format.
Also, it is necessary for this value to be UTF-8-encoded as well as URL-encoded before being set in the header.
A PUT operation using the X-Copy-From header is the same as performing a COPY operation.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

ETag
MD5 checksum of the request body. For example, an MD5 checksum of the files to be uploaded.
In order to perform full uploads, it is strongly recommended to calculate the MD5 checksum of the object in advance, set it for this header and issue the request. This value is not to be enclosed in quotation marks.
<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Content-Disposition**

Sets the browser behavior.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Content-Encoding**

Sets the Content-Encoding metadata.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Delete-At**

Specify the date on which an object should be deleted in UNIX Epoch timestamp format.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Delete-After**

Specify the number of seconds until an object is deleted. This value is converted to X-Delete-At in the object storage.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Object-Meta-name**

Object metadata {name} is the name of the metadata item.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..*</td>
</tr>
</tbody>
</table>

**If-None-Match**

Expect: Use in combination with 100-Continue. This is used to check if the server is holding a cache of the data already specified in the header.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Request parameters**

{account}
Name uniquely assigned by project
<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**{container}**  
Container name

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**{object}**  
Object name

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**signature**  
Uses a signature for requests in the URL feature with effective period.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**expires**  
Specifies the effective period of the signature in the URL feature with effective period.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**multipart-manifest**  
When ?multipart-manifest=put is specified, the object is uploaded as the manifest of a static large object. The request body includes a description of the manifest.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Response header**

**Content-Length**  
Number of bytes of the response body

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**ETag**  
When an object is smaller than 5 GB, this value will be the MD5 checksum of the uploaded object. This value is not to be enclosed in quotation marks.

When an ETag header is assigned and the PUT operation is successful for the request, this value will be the same as the specified ETag.
Also, when an ETag header has not been assigned, check if this value is the same as the ETag value of the object that was uploaded.

In the case of a static large object, a value with the retrieved MD5 checksum for a string that combines the ETag and MD5 checksums of each segment of the manifest is returned.

In the case of a dynamic large object, this value will be the MD5 checksum of a null character.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Content-Type**

MIME type of an object

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Trans-Id**

ID assigned to this request. This is used when inquiring about issues.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uuid</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Date**

Datetime when transaction was executed

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datetime</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Response elements**

None

**Example of request**

```
```

**Example of response**

```
HTTP/1.1 201 Created
Last-Modified: Fri, 17 Jan 2014 17:28:35 GMT
Content-Length: 116
Etag: d41d8cd98f00b204e9800998ecf8427e
Content-Type: text/html; charset=UTF-8
X-Trans-Id: tx4d5e4f06d3574e62bb732f-0052d96843
Date: Fri, 17 Jan 2014 17:28:35 GMT
```

2.7.4.11 Copy object COPY /v1/{account}/{container}/{object}

Copies an object.
This is the same as using PUT and specifying X-Copy-From.

When this operation is used to copy a manifest object, the new object becomes a normal object with all segments joined. Therefore, it is not possible to copy an object that has a combined size exceeding 5 GB.

The same metadata as that of the copy source object is assigned. If metadata is specified when making a request, the metadata of the copy destination object will be updated after copying is done.

When creation of an object is successful, the 201 Created status code is returned.

Caution: The object storage is in state in which a number of replicas exist, based on Eventual Consistency, and the COPY operation is performed with the latest replica selected. In other words, when COPY is used, this performs the same operation as when the X-Newest header is used.

**Request headers**

**X-Auth-Token**
Authentication token

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Destination**
Specify the copy destination container name and object name using the /{container}/{object} format. Also, it is necessary for the container name and object name to be UTF-8-encoded as well as URL-encoded before being set in the header.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Content-Type**
MIME type of an object

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Content-Encoding**
Sets the Content-Encoding metadata.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Content-Disposition**

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>
X-Object-Meta-name
Object metadata {name} is the name of the metadata item.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

Request parameters

{account}
Name uniquely assigned by project

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

{container}
Container name

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

{object}
Object name

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

Response Headers

X-Copied-From-Last-Modified
Last modified datetime of copy source object

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

X-Copied-From
Container name and object name of copy source object. This is returned using the {container}/
{object} format.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

Last-Modified
Datetime when an object was created, or the datetime when the metadata was modified

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

ETag
MD5 checksum of an object. This value is not to be enclosed in quotation marks.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Content-Type**

MIME type of an object

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Object-Meta-name**

Object metadata {name} is the name of the metadata item.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..</td>
</tr>
</tbody>
</table>

**X-Trans-Id**

ID assigned to this request. This is used when inquiring about issues.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uuid</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Date**

Datetime when transaction was executed

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datetime</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Response elements**

None

**COPY**

**Example of request**

```
curl -i $publicURL/marktwain/goodbye -X COPY -H "X-Auth-Token: $token" -H "Destination: janeausten/goodbye"
```

**Example of response**

```
HTTP/1.1 201 Created
Content-Length: 0
X-Copied-From-Last-Modified: Thu, 16 Jan 2014 21:19:45 GMT
X-Copied-From: marktwain/goodbye
Last-Modified: Fri, 17 Jan 2014 21:19:45 GMT
Etag: 451e372e48e0f6b1114fa0724aa79fa1
Content-Type: text/html; charset=UTF-8
X-Object-Meta-Movie: AmericanPie
X-Trans-Id: txdcb481ad49d24e9a81107-0052d97501
```
(Reference) When copying using a "X-Copy-From" header with PUT

Example of request

```bash
```

Example of response

```
HTTP/1.1 201 Created
Content-Length: 0
X-Copied-From-Last-Modified: Thu, 16 Jan 2014 21:19:45 GMT
X-Copied-From: marktwain/goodbye
Last-Modified: Fri, 17 Jan 2014 18:22:57 GMT
Etag: 451e372e48e0f6b1114fa0724aa79fa1
Content-Type: text/html; charset=UTF-8
X-Object-Meta-Movie: AmericanPie
X-Trans-Id: txdcb481ad49d24e9a81107-0052d97501
Date: Fri, 17 Jan 2014 18:22:57 GMT
```

2.7.4.12 Delete object DELETE /v1/{account}/{container}/{object}

Deletes an object

Objects are deleted immediately. If a GET, HEAD, POST, or DELETE operation is performed after deletion, the 404 Not Found error code is returned.

To delete static large objects, it is necessary to delete the manifest and split objects. By specifying the multipart-manifest=delete query parameter, the manifest and split objects are deleted at once.

Normally, the DELETE operation does not return the response body. When the multipart-manifest=delete query parameter is specified, a list of the manifests and split objects along with their deletion status will be included in the response body.

Response code when an error occurs: 400, 500, ...

Request header

**X-Auth-Token**

Authentication token If omitted, the request will fail, unless access using an access control list (ACL) is permitted by the account owner.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

Request parameters

**{account}**

Unique name of the account. An "account" is also referred to as a "project" or "tenant".
Unique name of the container.

Unique name of the object.

When the target is a static large object

- Specify `multipart-manifest=delete`
  The manifest and split objects will be deleted.
- Omit `multipart-manifest=delete`
  Only the manifest will be deleted. The split objects will not be deleted.

**Response headers**

**Content-Length**
When the operation is successful, this value will be 0. When the operation fails, this value will be the length of the error text stored in the response body.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Content-Type**
MIME type of the object.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Trans-Id**
ID assigned to this request. This is used when inquiring about issues.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uuid</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Date**
Datetime when transaction was executed.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datetime</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Response elements**
None

Delete the helloworld object from Marktwain content

**Example of request**
Example of response

HTTP/1.1 204 No Content
Content-Length: 0
Content-Type: text/html; charset=UTF-8
X-Trans-Id: tx36c7606fcd1843f59167c-0052d6fdac
Date: Wed, 15 Jan 2014 21:29:16 GMT

2.7.4.13 Retrieve object metadata HEAD /v1/{account}/
{container}/{object}

Retrieves object metadata
Response code when the state is normal: 204

Request headers
X-Auth-Token
Authentication token

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

X-Newest
When "True" is set for this header, a search of all replicas is performed, and the latest replica
is returned. If this header is omitted, the response will be faster because the search is for one
valid replica only. This item will increase the load on the system, so it should only be used when
absolutely necessary.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boolean</td>
<td>0..1</td>
</tr>
</tbody>
</table>

Request parameters
{account}
Unique name of the account. An "account" is also referred to as a "project" or "tenant".

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

{container}
Unique name of the container.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

{object}
Unique name of the object.
Data type | Cardinality
---|---
String | 1..1

**Query Parameters**

**Signature**
Uses a signature for requests in the URL feature with effective period.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Expires**
Specifies the effective period of the signature in the URL feature with effective period.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Response headers**

**Last-Modified**
Datetime when the object was created, or datetime when the metadata was updated previously.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Content-Length**
Size of the object.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Content-Type**
MIME type of the object.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**ETag**
For objects smaller than 5 GB, this value is the MD5 checksum of the object content.
When an object is smaller than 5 GB, the MD5 checksum of the object will be returned. This value is not to be enclosed in quotation marks.

In the case of a manifest object, a value with the retrieved MD5 checksum for a string that combines the ETag and MD5 checksums of each segment of the manifest is returned. This is not the MD5 checksum of the downloaded object. Also, the value is to be enclosed in double quotation marks.

You are strongly advised to calculate the MD5 checksum of the response body, and compare it with the value retrieved using the ETag header. If the values are different, this indicates that the contents are corrupt, so it will be necessary to try again.
Data type | Cardinality
--- | ---
String | 1..1

**Content-Encoding**
If a value has been set, the Content-Encoding metadata value will be returned. If a value has not been set, a value will not be returned by this operation.

Data type | Cardinality
--- | ---
String | 0..1

**Content-Disposition**
This is returned when the browser behavior has been set.
Refer to [http://www.ietf.org/rfc/rfc2183.txt](http://www.ietf.org/rfc/rfc2183.txt) for details on the value that is returned.

Data type | Cardinality
--- | ---
String | 0..1

**X-Delete-At**
The date on which an object will be deleted is returned in UNIX Epoch timestamp format. If omitted, this will not be returned.

Data type | Cardinality
--- | ---
Int | 0..1

**X-Object-Manifest**
Returns the dynamic large object settings. This value takes the container name and prefix name of the split object in the {container}/{prefix} format.

Data type | Cardinality
--- | ---
String | 0..1

**X-Object-Meta-name**
Object metadata {name} is the name of the metadata item.

Data type | Cardinality
--- | ---
String | 0..1

**X-Static-Large-Object**
If the object is the manifest object of a static large object, "True" will be returned.

Data type | Cardinality
--- | ---
String | 1..1

**X-Trans-Id**
ID assigned to this request. This is used when inquiring about issues.

Data type | Cardinality
--- | ---
Uuid | 1..1
**Date**
Datetime when transaction was executed

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datetime</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Response elements**
None.

**Retrieve object metadata**

**Example of request**

curl -i $publicURL/marktwain/goodbye -X HEAD -H "X-Auth-Token:$token"

**Example of response**

HTTP/1.1 200 OK
Content-Length: 14
Accept-Ranges: bytes
Last-Modified: Thu, 16 Jan 2014 21:12:31 GMT
Etag: 451e372e48e0f6b114fa0724aa79fa1
X-Timestamp: 1389906751.73463
X-Object-Meta-Book: GoodbyeColumbus
Content-Type: application/octet-stream
X-Trans-Id: tx37ea34dcd1ed40ca9bc7d-0052d04d6f
Date: Thu, 16 Jan 2014 21:13:19 GMT

2.7.4.14 Update object metadata POST /v1/{account}/
{container}/[object]

Creates or updates object metadata

Use the X-Object-Meta-{name} header to create or update custom metadata items. Specify the name of custom metadata item in {name}.

Previously assigned custom metadata items will be deleted. It will be necessary to use a POST request to re-create custom metadata items.

The system metadata will not be updated.

However, updating is possible using the following headers: Content-Type, Content-Encoding, Content-Disposition, and X-Delete-At. Unless these are specified as a request header, the metadata will not be changed.

Normal response codes: 202

**Request headers**

**X-Auth-Token**
Authentication token If omitted, the request will fail, unless access using an access control list (ACL) is permitted by the account owner.
<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Object-Meta-name**

Container metadata. `{name}` is the name of the metadata.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Delete-At**

Specify the date on which an object should be deleted in UNIX Epoch timestamp format.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Content-Disposition**

Sets the browser behavior.


<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Content-Encoding**

Specify the Content-Encoding metadata.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Delete-After**

Specify the time that should elapse until the object is deleted.

The Object Storage system holds this value as "X-Delete-At" metadata.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Content-Type**

Change the MIME type of an object.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**X-Detect-Content-Type**

Specify automatic detection of the Content-Type of an object.

When "True" is specified for this header, the value specified in the Content-Type header will be ignored, and the content type will be inferred from the file extension.
<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boolean</td>
<td>0..1</td>
</tr>
</tbody>
</table>

**Request parameters**

**{account}**
Unique name of the account. An "account" is also referred to as a "project" or "tenant".

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**{container}**
Unique name of the container.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**{object}**
Unique name of the object.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Response headers**

**Content-Length**
When the operation is successful, this value will be 0. When the operation fails, this value will be the length of the error text stored in the response body.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Content-Type**
MIME type of the object.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**X-Trans-Id**
ID assigned to this request. This is used when inquiring about issues.

<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uuid</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Date**
Datetime information of a transaction.
<table>
<thead>
<tr>
<th>Data type</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datetime</td>
<td>1..1</td>
</tr>
</tbody>
</table>

**Response elements**

The HTML content indicating that the metadata was created is returned.

**Create object metadata**

**Example of request**

```
curl -i $publicURL/marktwain/goodbye -X POST -H "X-Auth-Token: $token" -H "X-Object-Meta-Book: GoodbyeColumbus"
```

**Example of Response**

```html
HTTP/1.1 202 Accepted
Content-Length: 76
Content-Type: text/html; charset=UTF-8
X-Trans-Id: txb5fb5c91ba1f4f37bb648-0052d84b3f
Date: Thu, 16 Jan 2014 21:12:31 GMT
<html><h1>Accepted</h1><p>The request is accepted for processing.</p></html>
```

**Update object metadata**

**Example of request**

```
```

**Example of response**

```html
HTTP/1.1 202 Accepted
Content-Length: 76
Content-Type: text/html; charset=UTF-8
X-Trans-Id: tx5ec7ab81cd84ced887c8-0052d84ca4
Date: Thu, 16 Jan 2014 21:18:28 GMT
<html><h1>Accepted</h1><p>The request is accepted for processing.</p></html>
```