UpCloud API Documentation

API version 1.2.1
Updated 30 Apr 2015

1. INTRODUCTION

UpCloud is a cloud hosting platform used to run virtual servers. UpCloud provides:

- easy creation and hosting of virtual server instances
- management of hosting resources such as servers, storages and IP addresses efficiently
- quickly upscaling and downscaling servers
- migrating servers to different geographical locations
- a private network for communication between servers
- firewall services

The billing of resources is done by the hour on a utility basis.

This document describes the Application Programming Interface (API) to UpCloud. It gives information on how to use the API to build applications to control resources such as servers and storages on UpCloud. API client applications can be used to automate regular tasks on the cloud such as creating new servers and to abstract the end user from the cloud service.

This technical document is intended for developers building applications using the UpCloud API. Basic knowledge in programming is required to use the API. The API is independent from any programming language.

1.1. Version history

API version	Date	Note
1.0.0	Apr 27, 2012	First version of this document.
1.0.1	Jul 9, 2012	Added the error code CDROM_EJECT_FAILED to the Eject CD-ROM operation.
1.0.2	Oct 3, 2012	Added the new storage resource backup features.
1.0.3, 1.1.1	Aug 7, 2013	Added the Modify IP address operation for modifying PTR records. Storages are divided into two tiers: `hdd` and `ssd`. Modified server operations to support the new VNC connection method.
1.2.0	Mar 30, 2015	Storage tier `maxiops` added (replacing `ssd`). Added the `family` attribute to IP address and Firewall rule operations to support IPv6. Several other minor updates.
1.2.1	Apr 29, 2015	Server attribute `host` added for server details and `avoid_host` for more control on what host machine VMs should or should not reside on.

1.2. Changes from version 1.1

List of incompatible changes from version 1.1

- IP address and Firewall Rule operations now require `family` attribute.

- Customers can now configure servers to avoid certain hosts for added redundancy of environments.

2. ARCHITECTURE OVERVIEW

2.1. API overview

The UpCloud API consists of operations used to control resources on UpCloud. The API is a web service interface. HTTPS is used to connect to the API. The API follows the principles of a RESTful web service wherever possible. The base URL for all API operations is https://api.upcloud.com/. All API operations require authentication.

2.2. API versioning

The API uses versioning to allow backwards incompatible modifications in the service without affecting clients using older versions of the API.

The versioning uses an X.Y.Z scheme. The X.Y part is shown in the URL, e.g. https://api.upcloud.com/1.2/.

Backwards compatible changes in the API does not require changes to clients using the API. Such changes include but are not limited to adding or modifying error codes and messages or adding new features. When backwards compatible changes are made to the API, the Z version is incremented.

Small and large backwards incompatible changes increment the Y or X version respectively. Upgrading to a backwards incompatible API version require changing the URLs that are used to access the service.

Changes to the API are noted in the Version history.

If an invalid or inexistent API version is requested, the error code API_VERSION_INVALID is returned. If an obsolete API version is requested, the error code API_VERSION_OBSOLETE is returned.

2.3. Authentication

The UpCloud API uses HTTP Basic access authentication. An API user and a password is required to authenticate with the UpCloud API. The API user can be created in the UpCloud control panel.

Sample request header

```
GET /1.2/server HTTP/1.0
Host: api.upcloud.com
Authorization: Basic dXNlcm5hbWU6cGFzc3dvcmQ=
```

Normal response

If authentication succeeds, the API returns an HTTP 200 OK response. A sample response header for a successful authentication:

```
HTTP/1.0 200 OK
Date: Mon, 01 Jan 2012 00:00:00 GMT
Server: Apache
Content-Length: 39042
Connection: close
Content-Type: application/json; charset=UTF-8
```

Error response

If authentication fails, the API returns an HTTP 401 error response. An example response for a failed authentication attempt:

```
HTTP/1.0 401 Authorization Required
Date: Mon, 01 Jan 2012 00:00:00 GMT
Server: Apache
WWW-Authenticate: Basic, realm="API"
Content-Length: 157
Connection: close
Content-Type: application/json; Charset=UTF-8

{
    "error" : {
        "error_code" : "AUTHENTICATION_FAILED",
        "error_message" : "Authentication failed using the given username and password."
    }
}
```

2.4. Resource identifiers

UpCloud resources such as servers, storages and IP addresses are all referred to by unique identifiers. These identifiers are assigned when resources are created. Resources are always accessible by their unique identifiers as long as they exist on UpCloud.

The unique identifiers follow the version 4 format of the Universally unique identifier (UUID) standard.

Example UUID: *00104021-153d-4b0e-a4ec-730a5556442c*.

To help differentiate between various types of resources, different types of resources have different UUID prefixes:

Resource type	UUID prefix	UUID example
Servers	00	00ff2a9e-2e38-449e-8dee-7986a127d034
Storages	01	01fe1129-b1cd-48e4-a488-c51874082818

2.5. Request and response data formats

In addition to HTTP requests, many API operations include exchange of data in the request or response body. This data is encoded either in JavaScript Object Notation (JSON) or eXtensible Markup Language (XML). The default encoding in JSON.

All examples in this document are in JSON format.

The data type definition is given in the HTTP request header. The `Accept` header defines the data type for requests and the `Content-type` header defines the data type in responses.

It is possible to send data in one format and receive in the other format.

A sample HTTP request header

```
POST /1.2/server HTTP/1.0
Host: api.upcloud.com
Authorization: Basic <base64-encoded authentication string>
Accept: application/json
Content-Type: application/json
Content-Length: <length>
```

Valid data type definitions:
- *application/json*
- *application/xml*

Sample response in JSON format

```
HTTP/1.0 200 OK
Date: Tue, 03 Apr 2012 20:12:12 GMT
Server: Apache
Content-Length: 713
Connection: close
Content-Type: application/json; charset=UTF-8

{
   "servers" : {
      "server" : [
         {
            "core_number" : "1",
            "hostname" : "server1.example.com",
            "license" : 0,
            "memory_amount" : "2048",
            "state" : "started",
            "title" : "server1",
            "uuid" : "001c397f-9b24-42ce-9977-07f16a592fb3",
            "zone" : "fi-hel1"
         },
         {
            "core_number" : "0",
            "hostname" : "server2.example.com",
            "license" : 0,
            "memory_amount" : "512",
            "state" : "started",
            "title" : "server2",
            "uuid" : "0077fa3d-32db-4b09-9f5f-30d9e9afb316",
            "zone" : "fi-hel1"
         }
      ]
   }
}
```

Sample response in XML format

```
<?xml version="1.2" encoding="utf-8"?>
<servers>
  <server>
    <core_number>1</core_number>
    <hostname>server1.example.com</hostname>
    <license>0</license>
    <memory_amount>2048</memory_amount>
    <state>started</state>
    <title>server1</title>
    <uuid>001c397f-9b24-42ce-9977-07f16a592fb3</uuid>
    <zone>fi-hel1</zone>
  </server>
  <server>
    <core_number>0</core_number>
    <hostname>server2.example.com</hostname>
    <license>0</license>
    <memory_amount>512</memory_amount>
    <state>started</state>
    <title>server2</title>
    <uuid>0077fa3d-32db-4b09-9f5f-30d9e9afb316</uuid>
    <zone>fi-hel1</zone>
  </server>
</servers>
```

2.6. Response codes

API responses use regular HTTP codes to indicate successful and failed requests.

Successful requests

Successful requests are responded with a response code between 200 and 299. Possible response codes are:

HTTP status	Description
200 OK	The request was successful.
201 Created	The request resulted in the generation of a new resource such as a server or a storage.
202 Accepted	The request resulted in the asynchronous creation or update of an existing resource.

Failed requests

Failed requests include a machine readable error code and a human readable error description in the response body. Invalid requests are responded with a response code between 400 and 499. Possible response codes are:

HTTP status	Description
400 Bad request	The request has invalid, unknown or missing attributes or the request cannot be properly parsed.
401 Unauthorized	The request does not use or fails authentication.
402 Payment Required	The request cannot be processed because of insufficient credits.
403 Forbidden	The request accesses resources owned by some other user.
404 Not Found	The request accesses resources that do not exist, but may have existed before.
405 Method not allowed	The request uses an invalid HTTP verb, e.g. use $POST$ on an operation that only accepts GET.
406 Not Acceptable	The request has requirements given in the $Accept$ headers that the API service cannot comply with.
409 Conflict	The request cannot be processed because of a conflict between resources, e.g. trying to start an already started server.

API errors

Should there be a problem with the API service, a request is responded with an error code between 500 and 599. Possible response codes are:

HTTP status	Description
500 Internal Server Error	The request cannot be processed because of a temporary problem in the cloud service.
503 Service Unavailable	The API service is not in use because of maintenance.

The *error_code* parameter defines the more specific error code. The error codes are defined under each operation in this API documentation.

The *error_message* parameter gives a human-readable error description that can include dynamic content such as UUIDs of resources in order to help track down the source of the error.

Example error response

```
HTTP/1.0 404 Not Found
{
  "error" : {
    "error_message" : "The server 00af0f73-7082-4283-b925-811d1585774b does not
exist.",
    "error_code" : "SERVER_NOT_FOUND"
  }
}
```

3. ACCOUNTS

An API user account and password is required to access the UpCloud API. The API user is associated with an UpCloud account. A user account may have multiple API users to be used in different client software. In order to use resources from the cloud, the account must have enough credits.

3.1. Credits

Credits are used to pay for cloud resources such as servers, storages, network traffic and IP addresses. Credits are automatically deducted for used resources on an hourly basis. Should the user run out of credits, active resources will be disabled and reactivated after the credit balance returns to positive. Credits can be purchased from the UpCloud website.

3.2. Get account information

Returns information on the user's account.

Request

```
GET /1.2/account
```

Normal response

```
HTTP/1.0 200 OK
{
  "account" : {
     "credits" : "10000",
     "username" : "username"
  }
}
```

4. PRICING

UpCloud resources are billed by the hour according to the price list.

4.1. Licensing fees

In addition to resource prices, some resources have licensing fees. Licenses apply to commercial operating system software. The licensing fee is indicated by the `license` attribute of the storage resource. The total licensing fee of a single server is indicated by the server's `license` attribute, then multiplied by the number of used CPU cores. Using a shared CPU core is calculated as a half CPU core.

4.2. List prices

Returns a list resource prices.

Request

```
GET /1.2/price
```

Response

```
HTTP/1.0 200 OK
{
    "prices" : {
        "zone" : [
            {
                "firewall" : {
                    "amount" : 1,
                    "price" : 0.5
                },
                "io_request_backup" : {
                    "amount" : 1000000,
                    "price" : 10
                },
                "io_request_hdd" : {
                    "amount" : 1000000,
                    "price" : 0
                },
                "io_request_maxiops" : {
                    "amount" : 1000000,
                    "price" : 0
                },
                "io_request_ssd" : {
                    "amount" : 1000000,
                    "price" : 0
                },
                "ipv4_address" : {
                    "amount" : 1,
                    "price" : 0.3
                },
                "ipv6_address" : {
                    "amount" : 1,
                    "price" : 0
                },
                "name" : "fi-hel1",
                "public_ipv4_bandwidth_in" : {
                    "amount" : 1,
                    "price" : 0
                },
                "public_ipv4_bandwidth_out" : {
                    "amount" : 1,
                    "price" : 5
                },
                "public_ipv6_bandwidth_in" : {
                    "amount" : 1,
                    "price" : 0
                },
                "public_ipv6_bandwidth_out" : {
                    "amount" : 1,
                    "price" : 5
                },
                "server_core" : {
                    "amount" : 1,
                    "price" : 1.3
                },
                "server_memory" : {
                    "amount" : 256,
                    "price" : 0.45
                },
```

```
            "storage_backup" : {
               "amount" : 1,
               "price" : 0.007
            },
            "storage_hdd" : {
               "amount" : 1,
               "price" : 0.013
            },
            "storage_maxiops" : {
               "amount" : 1,
               "price" : 0.028
            },
            "storage_ssd" : {
               "amount" : 1,
               "price" : 0.05
            }
         }
      ]
   }
}
```

5. ZONES

A zone identifies the physical site where the cloud services are located. Different zones may be used to provide geographical and logical separation of servers.

All zones have their own storage network. Storages can only be attached to servers located in the same zone. In order to use storages from other zones, storages can be transferred between zones by cloning.

5.1. List available zones

Returns a list of available zones.
All servers and storages must set their zone attributes to one of the zone ids returned by this operation.

Request

```
GET /1.2/zone
```

Normal response

```
HTTP/1.0 200 OK
{
    "zones" : {
        "zone" : [
            {
                "id" : "fi-hel1",
                "description" : "Helsinki, Finland, zone 1"
            },
            {
                "id" : "uk-lon1",
                "description" : "London, United Kingdom, zone 1"
            },
            {
                "description" : "Chicago #1",
                "id" : "us-chi1"
            }
        ]
    }
}
```

6. TIMEZONES

A cloud server's hardware clock is set to a specific timezone. Some operating systems, such as Microsoft Windows, rely on the hardware clock being in local time. Others, such as Linux, use a default setting of a hardware clock in Coordinated Universal Time (UTC). The server's timezone setting can be adjusted to a local time if required by the operating system.

6.1. List timezones

Returns a list of available timezones. Timezones are used to set the hardware clock for servers.

The default server timezone is *UTC* unless configured to one of the timezone identifiers returned by this operation.

Request

```
GET /1.2/timezone
```

Normal response

```
HTTP/1.0 200 OK
{
    "timezones" : {
        "timezone" : [
            "Africa/Abidjan",
            "Africa/Accra",
            "Africa/Addis_Ababa",
            ...
            "Pacific/Truk",
            "Pacific/Wake",
            "Pacific/Wallis",
            "UTC"
        ]
    }
}
```

7. SERVERS

This chapter describes operations for creating and managing servers on UpCloud. A server is a single virtual machine instance running on UpCloud.

The server configuration defines which storage devices the server is attached to, which IP addresses can be used and how the server can be reached for remote management. A server must have at least one storage device attached in order to be started. Servers may have from zero to five public IPv4 and IPv6 addresses. All servers have a private IP address that cannot be removed.

7.1. Server states

The server state indicates the server's current status.

State	Description
started	The server is running.
stopped	The server is stopped.
maintenance	The server is in maintenance mode.
error	The server has encountered an error. This means the server is inaccessible due to an error in the cloud service.

The normal server states are *started* and *stopped*. The maintenance state is a temporary state used when the cloud service updates the server. If the server is in *error* state, it will return automatically in the *started* or *stopped* state when the issue is resolved.

7.2. List server configurations

Returns a list of available server configurations. A server configuration consists of a combination of CPU core count and main memory amount. All servers are created using these configurations.

All UpCloud servers have dedicated virtual CPU cores in order to provide maximal and predictable performance.

Request

```
GET /1.2/server_size
```

Normal response

```
HTTP/1.0 200 OK
{
    "server_sizes" : {
        "server_size" : [
            {
                "core_number" : "1",
                "memory_amount" : "512"
            },
            {
                "core_number" : "1",
                "memory_amount" : "768"
            },
            ... many more combinations ...
            {
                "core_number" : "10",
                "memory_amount" : "65024"
            },
            {
                "core_number" : "10",
                "memory_amount" : "65536"
            }
        ]
    }
}
```

Normal response

7.3. List servers

Returns a list of all servers associated with the current account.

Only the servers' most relevant information is returned by this operation. Further details on individual servers can be requested with the Get server details operation.

Request

```
GET /1.2/server
```

Normal response

```
HTTP/1.0 200 OK
{
  "servers" : {
    "server" : [
      {
        "zone" : "fi-hel1",
        "core_number" : "1",
        "title" : "Helsinki server",
        "hostname" : "fi.example.com",
        "memory_amount" : "512",
        "uuid" : "00798b85-efdc-41ca-8021-f6ef457b8531",
        "state" : "started"
      },
      {
        "zone" : "uk-lon1",
        "core_number" : "1",
        "title" : "London server",
        "hostname" : "uk.example.com",
        "memory_amount" : "512",
        "uuid" : "009d64ef-31d1-4684-a26b-c86c955cbf46",
        "state" : "stopped"
      }
    ]
  }
}
```

7.4. Get server details

Returns detailed information about a specific server.

Request

```
GET /1.2/server/00798b85-efdc-41ca-8021-f6ef457b8531
```

Normal response

```
HTTP/1.0 200 OK
{
    "server" : {
        "boot_order" : "disk",
        "core_number" : "0",
        "firewall" : "on",
        "host" : 7653311107,
        "hostname" : "server1.example.com",
        "ip_addresses" : {
            "ip_address" : [
                {
                    "access" : "private",
                    "address" : "10.0.0.0",
                    "family" : "IPv4"
                },
                {
                    "access" : "public",
                    "address" : "0.0.0.0",
                    "family" : "IPv4"
                },
                {
                    "access" : "public",
                    "address" : "xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx",
                    "family" : "IPv6"
                }
            ]
        },
        "license" : 0,
        "memory_amount" : "512",
        "nic_model" : "virtio",
        "state" : "started",
        "storage_devices" : {
            "storage_device" : [
                {
                    "address" : "virtio:0",
                    "storage" : "012580a1-32a1-466e-a323-689ca16f2d43",
                    "storage_size" : 100,
                    "storage_title" : "Storage for server1.example.com",
                    "type" : "disk"
                }
            ]
        },
        "timezone" : "UTC",
        "title" : "server1.example.com",
        "uuid" : "0077fa3d-32db-4b09-9f5f-30d9e9afb565",
        "video_model" : "cirrus",
        "vnc" : "on",
        "vnc_host" : "fi-he11.vnc.upcloud.com",
        "vnc_password" : "aabbccdd",
        "vnc_port" : "00000",
        "zone" : "fi-hel1"
    }
}
```

The *license* attribute indicates the amount of credits per hour per CPU core required by the server license. The server's *license* attribute is the sum of all the attached storages' *license* attributes.

7.5. Create server

Creates a new server instance.

There are three ways to create a server:

- from a template,
- by cloning another server, or,
- by installing the server from scratch using an operating system installation media.

New servers are automatically started. By default, the server is accessible using VNC. A VNC password is generated and returned in the output of the create server operation.

Creating from a template

UpCloud provides templates for various operating systems. Servers created from templates are preconfigured. The configurations vary between templates and may include, for example, setting the hostname, resizing the filesystem to fill the storage size and setting a root or other administrative account password.

A list of available templates can be obtained with the following request:

```
GET /1.2/storage/template
```

See List storages for more information on storage resource listings.

When creating a server from a template, the root password is returned in the output of the operation. This password is not stored anywhere and cannot be retrieved later. For security reasons, the password should be changed as soon as possible.

Creating a server from a template is an asynchronous operation. While the create operation is running, the server is in *maintenance* state as the server is being configured. As soon as the server is ready, the state will change to *started*.

It is currently not possible for users to create templates.

Request

```
POST /1.2/server
{
  "server" : {
    "zone" : "fi-hel1",
    "title" : "My Debian server",
    "hostname" : "debian.example.com",
    "core_number" : "4",
    "memory_amount" : "8192",
    "storage_devices" : {
      "storage_device" : [
        {
          "action" : "clone",
          "storage" : "01000000-0000-4000-8000-000020010600",
          "title" : "Debian from a template"
        }
      ]
    }
  }
}
```

The storage size is initially that of the template. When the server creation is complete, the storage will be resized and the size attribute will be set accordingly.

Cloning another server

An exact copy of another server can be obtained by cloning the server's storage devices. Cloning is a convenient method of creating multiple identical server instances.

The state of the cloned storage must be *online*.

Request

```
POST /1.2/server
{
  "server" : {
    "zone" : "fi-hel1",
    "title" : "Another Debian server",
    "hostname" : "debian2.example.com",
    "core_number" : "4",
    "memory_amount" : "8192",
    "storage_devices" : {
      "storage_device" : [
        {
          "action" : "clone",
          "storage" : "0169b4f8-051c-4a86-9484-f5b798249949",
          "title" : "Storage for another Debian server"
        }
      ]
    }
  }
}
```

Installing a server from scratch

Attribute	Accepted values	Default value	Required	Description
boot_order	`disk` / `cdrom` / `disk,cdrom` / `cdrom,disk`	`disk`	no	The storage device boot order.
core_number	A valid combination with `memory_amount`	smallest possible	no	The number of CPU cores dedicated to the server. See List server configurations.
firewall	`on` / `off`	`on`	no	The state of the server firewall rules.
hostname	A valid hostname		yes	A valid domain name, e.g. `host.example.com`. The maximum length is 128 characters.
avoid_host	A valid host id		no	Use this to make sure VMs do not reside on specific host. Refers to value from host -attribute. Useful when building HA-environments.
ip_addresses	1-10 `ip_address` blocks.	One IPv4 and one IPv6 address block	no	The `ip_addresses` block contains `ip_address` blocks that define the network interfaces.
memory_amount	A valid combination with `core_number`	smallest possible	no	The amount of main memory in megabytes. See List server configurations.
nic_model	`e1000` / `virtio` / `rtl8139`	`e1000`	no	The model of the server's network interfaces.
password_delivery	`none` / `email` / `sms`	`email`	no	The delivery method for the server's root password.
storage_devices	1-4 `storage_device` blocks		yes	The `storage_devices` block contains `storage_device` blocks that define the attached storages.
timezone	A valid timezone identifier	UTC	no	A timezone identifier, e.g. `Europe/Helsinki`. See Timezones.
title	0-64 characters		yes	A short description.
video_model	`vga` / `cirrus`	`vga`	no	The model of the server's video interface.
vnc	on / off	off	no	Is VNC service enabled.
vnc_password	8-32 characters of a-z, A-Z and 0-9.	randomly generated	no	The VNC remote access password.
zone	A valid zone identifier		yes	The zone in which the server will be hosted, e.g. `fi-hel1`. See Zones.

The server's operating system can be installed from a CD-ROM by attaching an empty storage device and loading an operating system installation CD-ROM.

UpCloud provides a variety of installation CD-ROMs for different operating systems. A list of CD-ROMs can be retrieved using the following request:

```
GET /1.2/storage/cdrom
```

See List storages for more information on storage resource listings.
VNC is used to connect to the server during installation.

Request

```
POST /1.2/server
{
  "server" : {
    "zone" : "fi-hel1",
    "title" : "My Debian server",
    "hostname" : "debian.example.com",
    "avoid_host" : "7653311107";
    "core_number" : "4",
    "memory_amount" : "8192",
    "storage_devices" : {
      "storage_device" : [
        {
          "action" : "create",
          "size" : "20",
          "tier" : "maxiops",
          "title" : "Debian from scratch"
        },
        {
          "action" : "attach",
          "storage" : "01000000-0000-4000-8000-000020010301",
          "type" : "cdrom"
        },
      ]
    }
    "ip_addresses" : {
        "ip_address" : [
          { "access" : "private", "family" : "IPv4" },
          { "access" : "public", "family" : "IPv4" },
          { "access" : "public", "family" : "IPv6" }
        ]
    }
  }
}
```

Attributes

The `storage_devices` block may contain 1-4 storage devices. Each block can have the following attributes:

Attribute	Accepted values	Default value	Required	Description
action	`create` / `clone` / `attach`		yes	The method used to create or attach the specified storage.
address	`ide:[01]:[01]` / `scsi:0:[0-7]` / `virtio:[0-7]`	The next available	no	The device address the storage will be attached to.
size	10-1024	if `action` is `clone`, same as original	yes if `action` is `create`	The size of the storage device in gigabytes. This attribute is applicable only if action is `create` or `clone`.
storage	A valid storage UUID		yes if `action` is `clone` or `attach`	The UUID of the storage device to be attached or cloned. Applicable only if `action` is `attach` or `clone`.
tier	`hdd` / `ssd`	`hdd`	No	The storage tier to use. See [Storage tiers](#).
title	0-64 characters		no	A short description for the storage.
type	`disk` / `cdrom`	`disk`	no	The device type the storage will be attached as. See [Storage types](#).

The ip_addresses block may contain 1-11 IP addresses. Each block can have the following attributes:

Attribute	Accepted values	Default value	Required	Description
access	`private` / `public`	public	yes	Is address for private or public network.
family	`IPv4` / `IPv6`	IPv4	no	The address family for new address. Private addresses can be only IPv4.

Normal response

```
HTTP/1.0 202 Accepted
{
   "server" : {
      "boot_order" : "disk",
      "core_number" : "4",
      "firewall" : "on",
      "hostname" : "debian.example.com",
      "ip_addresses" : {
         "ip_address" : [
            {
               "access" : "private",
               "address" : "10.0.0.0",
               "family" : "IPv4"
            },
            {
               "access" : "public",
               "address" : "x.x.x.x",
               "family" : "IPv4"
            },
            {
               "access" : "public",
               "address" : "xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx",
               "family" : "IPv6"
            }
         ]
      },
      "license" : 0,
      "memory_amount" : "8192",
      "nic_model" : "e1000",
      "password" : "aabbccdd",
      "state" : "maintenance",
      "storage_devices" : {
         "storage_device" : [
            {
               "address" : "virtio:0",
               "storage" : "0169b4f8-051c-4a86-9484-f5b798249949",
               "storage_size" : 1,
               "storage_title" : "Debian from a template",
               "type" : "disk"
            }
         ]
      },
      "timezone" : "UTC",
      "title" : "My Debian server",
      "username" : "root",
      "uuid" : "00c78863-db86-44ea-af70-d6edc4d162bf",
      "video_model" : "cirrus",
      "vnc" : "off",
      "vnc_password" : "aabbccdd",
      "zone" : "fi-hel1"
   }
}
```

Error responses

HTTP status	Error code	Description
400 Bad Request	ACTION_INVALID	The attribute *action* has an invalid value.
400 Bad Request	ACTION_MISSING	The required attribute *action* is missing from the request.
400 Bad Request	BOOT_ORDER_INVALID	The attribute *boot_order* has an invalid value.
400 Bad Request	CORE_MEMORY_UNSUPPORTED	The combination of *core_number* and *memory_amount* is not supported. See List server configurations.
400 Bad Request	FIREWALL_INVALID	The attribute *firewall* has an invalid value.
400 Bad Request	CORE_NUMBER_INVALID	The attribute *core_number* has an invalid value.
400 Bad Request	HOSTNAME_INVALID	The attribute *hostname* has an invalid value. Value is case sensitive and must be in lower case.
400 Bad Request	HOSTNAME_MISSING	The required attribute *hostname* is missing from the request.
400 Bad Request	MEMORY_AMOUNT_INVALID	The attribute *memory_amount* has an invalid value.
400 Bad Request	NIC_MODEL_INVALID	The attribute *nic_model* has an invalid value.
400 Bad Request	PASSWORD_DELIVERY_INVALID	The attribute *password_delivery* has an invalid value.
400 Bad Request	SERVER_TITLE_INVALID	The attribute *title* in the server block has an invalid value.
400 Bad Request	SERVER_TITLE_MISSING	The required attribute *title* is missing from the *server* block.
400 Bad Request	SIZE_INVALID	The attribute *size* has an invalid value. If *action* is *clone*, *size* must be greater than the size of the cloned storage.
400 Bad Request	SIZE_MISSING	The required attribute *size* is missing from the request.
400 Bad Request	STORAGE_DEVICE_INVALID	A malformed *storage_device* block.
400 Bad Request	STORAGE_DEVICE_MISSING	The *storage_device* block is missing from the request.
400 Bad Request	STORAGE_DEVICES_INVALID	A malformed *storage_devices* block.
400 Bad Request	STORAGE_DEVICES_MISSING	The *storage_devices* block is missing from the request.

HTTP status	Error code	Description
400 Bad Request	STORAGE_INVALID	The attribute *storage* has an invalid value.
400 Bad Request	STORAGE_MISSING	The required attribute *storage* is missing from a `storage_device` block.
400 Bad Request	STORAGE_TITLE_INVALID	The attribute *title* in a storage_device block has an invalid value.
400 Bad Request	STORAGE_TITLE_MISSING	The required attribute *title* is missing from the *storage_device* block.
400 Bad Request	TIMEZONE_INVALID	The attribute *timezone* has an invalid value.
400 Bad Request	TYPE_INVALID	The attribute *type* has an invalid value.
400 Bad Request	TIER_INVALID	The attribute *tier* has an invalid value.
400 Bad Request	VIDEO_MODEL_INVALID	The attribute *video_model* has an invalid value.
400 Bad Request	VNC_INVALID	The attribute *vnc* has an invalid value.
400 Bad Request	VNC_PASSWORD_INVALID	The attribute *vnc_password* has an invalid value.
400 Bad Request	ZONE_INVALID	The attribute *zone* has an invalid value.
400 Bad Request	ZONE_MISSING	The required attribute *zone* is missing from the request.
402 Payment Required	INSUFFICIENT_CREDITS	There are not enough credits to perform the requested action. See [Credits](#).
403 Forbidden	STORAGE_FORBIDDEN	The storage exists, but is owned by another account.
404 Not Found	STORAGE_NOT_FOUND	The storage does not exist.
404 Not Found	ZONE_NOT_FOUND	The zone does not exist.
409 Conflict	CDROM_DEVICE_IN_USE	The request has multiple `storage_device` blocks that attach storages as a CD-ROM. Only one CD-ROM device can be attached per server.
409 Conflict	DEVICE_ADDRESS_IN_USE	The request has multiple `storage_device` blocks attached to the same device address.
409 Conflict	IP_ADDRESS_RESOURCES_UNAVAILABLE	There were not enough IP addresses available in the specified zone to create the requested server.
409 Conflict	MULTIPLE_TEMPLATES	The request has multiple *storage_device* blocks referring to templates. Only one storage template can be used to create a server.

HTTP status	Error code	Description
409 Conflict	PUBLIC_STORAGE_ATTACH	Attaching a storage of access type *public* is not allowed. See [Storage access types]().
409 Conflict	SERVER_RESOURCES_UNAVAILABLE	There are not enough server resources available in the specified zone to create the requested server.
409 Conflict	STORAGE_ATTACHED_AS_CDROM	A storage resource to be attached as a disk is already attached to some server as a CD-ROM.
409 Conflict	STORAGE_ATTACHED_AS_DISK	A storage resource to be attached as a CD-ROM is already attached to some server as a disk.
409 Conflict	STORAGE_DEVICE_LIMIT_REACHED	The request contained more than four *storage_device* blocks.
409 Conflict	STORAGE_IN_USE	The request has multiple *storage_device* blocks referring to the same storage resource.
409 Conflict	STORAGE_RESOURCES_UNAVAILABLE	There are not enough storage resources available in the specified zone to create the requested storages.
409 Conflict	STORAGE_STATE_ILLEGAL	The storage is in a state in which it cannot be used. See [Storage states]().
409 Conflict	STORAGE_TYPE_ILLEGAL	The type of the storage to be attached is illegal. Attaching a cdrom or a template as a disk is not possible. See [Storage types]().
409 Conflict	ZONE_MISMATCH	The storage is located in a different zone than the one the server is created in.

7.6. Start server

Starts a stopped server. The server state must be *stopped*.

Example request

```
POST /1.2/server/00c78863-db86-44ea-af70-d6edc4d162bf/start
```

Normal response

```
HTTP/1.0 200 OK
{
   "server" : {
      "boot_order" : "disk",
      "core_number" : "4",
      "firewall" : "on",
      "hostname" : "debian.example.com",
      "avoid_host" : "7653311107";
      "ip_addresses" : {
         "ip_address" : [
            {
               "access" : "private",
               "address" : "10.0.0.0",
               "family" : "IPv4"
            },
            {
               "access" : "public",
               "address" : "x.x.x.x",
               "family" : "IPv4"
            },
            {
               "access" : "public",
               "address" : "xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx",
               "family" : "IPv6"
            }
         ]
      },
      "license" : 0,
      "memory_amount" : "8192",
      "nic_model" : "virtio",
      "password" : "aabbccdd",
      "state" : "started",
      "storage_devices" : {
         "storage_device" : [
            {
               "address" : "virtio:0",
               "storage" : "0169b4f8-051c-4a86-9484-f5b798249949",
               "storage_size" : "10",
               "storage_title" : "Debian from a template",
               "type" : "disk"
            }
         ]
      },
      "timezone" : "UTC",
      "title" : "My Debian server",
      "username" : "root",
      "uuid" : "00c78863-db86-44ea-af70-d6edc4d162bf",
      "video_model" : "cirrus",
      "vnc" : "off",
      "vnc_password" : "aabbccdd",
      "zone" : "fi-hel1"
   }
}
```

Error responses

HTTP status	Error code	Description
400 Bad Request	SERVER_INVALID	The server UUID has an invalid value.
404 Not Found	SERVER_NOT_FOUND	The server does not exist.
402 Payment Required	INSUFFICIENT_CREDITS	There are not enough credits to perform the requested action. See Credits.
403 Forbidden	SERVER_FORBIDDEN	The server exists, but is owned by another account.
409 Conflict	NO_STORAGES_ATTACHED	There are no storage devices currently attached to the server. A server cannot be started with no storage devices.
409 Conflict	SERVER_RESOURCES_UNAVAILABLE	There are not enough server resources available in the specified zone to start the requested server.
409 Conflict	SERVER_STATE_ILLEGAL	The server is in a state in which it cannot be used. See Server states.

7.7. Stop server

Stops a started server.

The server state must be *started*.

Stopping a server can be done in three different ways:

- by hard stopping, or,
- soft stopping with a timeout, or,
- soft stopping without a timeout.

Hard stopping a server is the equivalent of physically unplugging the server.

Soft stopping a server sends an ACPI signal to the server. If a timeout is set, the timeout period has passed and the server is still running, a hard stop is performed. If no timeout is set, only the ACPI signal is sent.

Request

```
POST /1.2/server/00c78863-db86-44ea-af70-d6edc4d162bf/stop
{
  "stop_server" : {
    "stop_type" : "soft",
    "timeout" : "60"
  }
}
```

Attributes

Attribute	Accepted values	Default value	Required	Description
stop_type	*soft* / *hard*	*soft*	no	Type of stop operation performed on the server.
timeout	*1-600*		no	The stop timeout in seconds.

Normal response

```
HTTP/1.0 200 OK
{
   "server" : {
      "boot_order" : "disk",
      "core_number" : "4",
      "firewall" : "on",
      "hostname" : "debian.example.com",
      "ip_addresses" : {
         "ip_address" : [
            {
               "access" : "private",
               "address" : "10.0.0.0",
               "family" : "IPv4"
            },
            {
               "access" : "public",
               "address" : "x.x.x.x",
               "family" : "IPv4"
            },
            {
               "access" : "public",
               "address" : "xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx",
               "family" : "IPv6"
            }
         ]
      },
      "license" : 0,
      "memory_amount" : "8192",
      "nic_model" : "e1000",
      "password" : "aabbccdd",
      "state" : "started",
      "storage_devices" : {
         "storage_device" : [
            {
               "address" : "virtio:0",
               "storage" : "0169b4f8-051c-4a86-9484-f5b798249949",
               "storage_size" : "10",
               "storage_title" : "Debian from a template",
               "type" : "disk"
            }
         ]
      },
      "timezone" : "UTC",
      "title" : "My Debian server",
      "username" : "root",
      "uuid" : "00c78863-db86-44ea-af70-d6edc4d162bf",
      "video_model" : "cirrus",
      "vnc" : "off",
      "vnc_password" : "aabbccdd",
      "zone" : "fi-hel1"
   }
}
```

Note: the server's state is still *started* after the operation. The state will change to *stopped* once the server has shut down.

Error responses

HTTP status	Error code	Description
400 Bad Request	STOP_TYPE_INVALID	The attribute *stop_type* has an invalid value.
400 Bad Request	TIMEOUT_INVALID	The attribute *timeout* has an invalid value.
400 Bad Request	SERVER_INVALID	The *server* UUID has an invalid value.
403 Forbidden	SERVER_FORBIDDEN	The server exists, but is owned by another account.
404 Not Found	SERVER_NOT_FOUND	The server does not exist.
409 Conflict	SERVER_STATE_ILLEGAL	The server is in a state in which it cannot be used. See Server states.

7.8. Restart server

Stops and starts a *server*.

The server state must be *started*.

A restart can be done in three ways:

- by hard stopping, or,
- by soft stopping with a timeout, or,
- by soft stopping without a timeout.

Hard stopping the server in the equivalent of power cycling the server. The server is then started.

Soft stopping sends an ACPI signal to the server. The API then waits for the server to shut down before starting it. If the server has not shut down within the *timeout* period, the action indicated by *timeout_action* is performed. A value of *destroy* hard stops the server which is then started. A value of *ignore* stops the operation and the server will not be started.

Request

```
POST /1.2/server/00c78863-db86-44ea-af70-d6edc4d162bf/restart
{
  "restart_server" : {
    "stop_type" : "soft",
    "timeout" : "60",
    "timeout_action" : "destroy"
  }
}
```

Attributes

Attributes	Accepted values	Default value	Required	Description
stop_type	*soft* / *hard*	*soft*	no	Restart type
timeout	1-600		yes if *type* is *soft*	Stop timeout in seconds
timeout_action	*destroy* / *ignore*	*ignore*	no	Action to take if timeout limit is exceeded.

Normal response

```
HTTP/1.0 200 OK
{
   "server" : {
      "boot_order" : "disk",
      "core_number" : "4",
      "firewall" : "on",
      "hostname" : "debian.example.com",
      "ip_addresses" : {
         "ip_address" : [
            {
               "access" : "private",
               "address" : "10.0.0.0",
               "family" : "IPv4"
            },
            {
               "access" : "public",
               "address" : "x.x.x.x",
               "family" : "IPv4"
            },
            {
               "access" : "public",
               "address" : "xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx",
               "family" : "IPv6"
            }
         ]
      },
      "license" : 0,
      "memory_amount" : "8192",
      "nic_model" : "e1000",
      "state" : "started",
      "storage_devices" : {
         "storage_device" : [
            {
               "address" : "virtio:0",
               "storage" : "0169b4f8-051c-4a86-9484-f5b632149949",
               "storage_size" : 10,
               "storage_title" : "Debian from a template",
               "type" : "disk"
            }
         ]
      },
      "timezone" : "UTC",
      "title" : "My Debian server",
      "uuid" : "00c96963-db86-44ea-af70-d6edc4d162bf",
      "video_model" : "cirrus",
      "vnc" : "off",
      "vnc_password" : "aabbccdd",
      "zone" : "fi-hel1"
   }
}
```

Error responses

HTTP status	Error code	Description
400 Bad Request	STOP_TYPE_INVALID	The attribute stop_type has an invalid value.
400 Bad Request	TIMEOUT_MISSING	The required attribute timeout is missing from the request.
400 Bad Request	TIMEOUT_INVALID	The attribute timeout has an invalid value.
400 Bad Request	TIMEOUT_ACTION_INVALID	The attribute timeout_action has an invalid value.
400 Bad Request	SERVER_INVALID	The server UUID has an invalid value.
403 Forbidden	SERVER_FORBIDDEN	The server exists, but is owned by another account.
404 Not Found	SERVER_NOT_FOUND	The server does not exist.
409 Conflict	SERVER_STATE_ILLEGAL	The server is in a state in which it cannot be used. See Server states.

7.9. Modify server

Modifies the configuration of an existing server.

Attaching and detaching storages as well as assigning and releasing IP addresses have their own separate operations. See Attach storage, Detach storage, Assign IP address and Release IP address.

In order to change the *core_number* or *memory_amount* attributes, the server state must be *stopped*.

Request

```
PUT /1.2/server/00c78863-db86-44ea-af70-d6edc4d162bf
{
  "server" : {
    "core_number" : "8",
    "memory_amount" : "16384"
  }
}
```

Attributes

Attribute	Accepted values	Default value	Required	Description
boot_order	*disk* / *cdrom* / *disk,cdrom* / *cdrom,disk*		no	The storage device boot order.
core_number	A valid combination with *memory_amount*		no	The number of CPU cores dedicated to the server. See List server configurations.
firewall	*on* / *off*	*on*	no	The state of the server firewall rules.
hostname	A valid hostname		no	A valid hostname, e.g. host.example.com. The maximum length is 128 characters.
memory_amount	A valid combination with *core_number*		no	The amount of main memory in megabytes. See List server configurations.
nic_model	*e1000* / *virtio* / *rtl8139*	*e1000*	no	The model of the server's network interfaces.
title	0-64 characters		no	A short description.
timezone	A valid timezone identifier		no	A timezone identifier, e.g. *Europe/ Helsinki*. See Timezones.
video_model	*vga* / *cirrus*	*vga*	no	The model of the server's video interface.
vnc	*on* / *off*	off	no	The state of the VNC remote access service.
vnc_password	8-32 characters of a-z, A-Z and 0-9.		no	The VNC remote access password.

Normal response

```
HTTP/1.0 202 Accepted
{
    "server" : {
        "boot_order" : "disk",
        "core_number" : "8",
        "firewall" : "on",
        "hostname" : "debian.example.com",
        "ip_addresses" : {
                {
                    "access" : "private",
                    "address" : "10.0.0.0",
                    "family" : "IPv4"
                },
                {
                    "access" : "public",
                    "address" : "x.x.x.x",
                    "family" : "IPv4"
                },
                {
                    "access" : "public",
                    "address" : "xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx",
                    "family" : "IPv6"
                }
            ]
        },
        "license" : 0,
        "memory_amount" : "16384",
        "nic_model" : "virtio",
        "password" : "aabbccdd",
        "state" : "started",
        "storage_devices" : {
            "storage_device" : [
                {
                    "address" : "virtio:0",
                    "storage" : "0169b4f8-051c-4a86-9484-f5b798249949",
                    "storage_size" : "10",
                    "storage_title" : "Debian from a template",
                    "type" : "disk"
                }
            ]
        },
        "timezone" : "UTC",
        "title" : "My Debian server",
        "username" : "root",
        "uuid" : "00c78863-db86-44ea-af70-d6edc4d162bf",
        "video_model" : "cirrus",
        "vnc: " : "off",
        "vnc_password" : "aabbccdd",
        "zone" : "fi-hel1"
    }
}
```

Error responses

HTTP status	Error code	Description
400 Bad Request	BOOT_ORDER_INVALID	The attribute *boot_order* has an invalid value.
400 Bad Request	CORE_MEMORY_UNSUPPORTED	The combination of *core_number* and *memory_amount* is not supported. See List server configurations.
400 Bad Request	CORE_NUMBER_INVALID	The attribute *core_number* has an invalid value.
400 Bad Request	FIREWALL_INVALID	The attribute *firewall* has an invalid value.
400 Bad Request	HOSTNAME_INVALID	The attribute *hostname* has an invalid value.
400 Bad Request	MEMORY_AMOUNT_INVALID	The attribute *memory_amount* has an invalid value.
400 Bad Request	NIC_MODEL_INVALID	The attribute *nic_model* has an invalid value.
400 Bad Request	SERVER_INVALID	The server UUID has an invalid value.
400 Bad Request	TITLE_INVALID	The attribute *title* has an invalid value.
400 Bad Request	TIMEZONE_INVALID	The attribute *timezone* has an invalid value.
400 Bad Request	VIDEO_MODEL_INVALID	The attribute *video_model* has an invalid value.
400 Bad Request	VNC_INVALID	The attribute *vnc* has an invalid value.
400 Bad Request	VNC_PASSWORD_INVALID	The attribute *vnc_password* has an invalid value.
403 Forbidden	SERVER_FORBIDDEN	The server exists, but is owned by another account.
404 Not Found	SERVER_NOT_FOUND	The server does not exist.
409 Conflict	SERVER_STATE_ILLEGAL	The server is in a state in which it cannot be used. See Server states.

7.10. Delete server

Deletes a server.

The server state must be *stopped*. Storage devices attached to the server are automatically detached and can be reattached to other servers. IP addresses used by the server are released.

Request

```
DELETE /1.2/server/00c78863-db86-44ea-af70-d6edc4d162bf
```

Normal response

```
HTTP/1.0 204 No Content
```

Error responses

HTTP status	Error code	Description
400 Bad Request	SERVER_INVALID	The server UUID has an invalid value.
403 Forbidden	SERVER_FORBIDDEN	The server exists, but is owned by another account.
404 Not Found	SERVER_NOT_FOUND	The server does not exist.
409 Conflict	SERVER_STATE_ILLEGAL	The server is in a state in which it cannot be used. See Server states.
409 Conflict	STORAGE_STATE_ILLEGAL	One of the attached storages is in a state in which it cannot be used. See Storage states.

8. STORAGES

A storage device is a block device similar to a physical hard disk. It can be used to install an operating system or hold some other filesystem for a server.

Storages can be freely attached to any server within the same zone and under the same account. A maximum of four storage devices can be attached to a server at the same time.

8.1. Storage types

There are four different storage types: *disk*, *cdrom*, *template* and *backup*.

Normal storages

Normal storage resources are used to store operating system and application data. This is the only user writeable storage type.

CD-ROMs

CD-ROM resources are used as a read-only media, typically for server installations or crash recovery.

Templates

Templates are special storage resources which are used to create new servers with a preconfigured operating system.

Backups

Backups are storages containing a point-in-time backup of a normal storage. Data on a normal storage can be restored from one of its backups. Backups can also be cloned to create a new normal storage resource. Backups can be created manually or automatically using backup rules.

8.2. Storage tiers

Storage resources are divided into two tiers: *hdd* and *maxiops*. Storage tiers affect both the performance and price of the storage.

HDD storages

Data is stored on hard disks resulting both in low-costs and high performance.

MaxIOPS storages

Data is stored on MaxIOPS storage arrays resulting in highest throughput and lowest response times.

8.3. Storage access types

Storage access types are *public* and *private*.

Public storages

Public storages are visible to all users. Public storages include CD-ROM images and templates. CD-ROM images can be used to install operating systems and to rescue unbootable systems. Templates are used to create servers with a preconfigured operating system.

Private storages

Private storages are visible only to the specific user account. Users can only create private storages.

8.4. Storage states

The storage state indicates the storage's current status.

State	Description
online	The storage resource is ready for use. The storage can be attached or detached.
maintenance	Maintenance work is currently performed on the storage. The storage may have been newly created or it is being updated by some API operation.
cloning	The storage resource is currently the clone source for another storage.
backuping	The storage resource is currently being backed up to another storage.
error	The storage has encountered an error and is currently inaccessible.

8.5. Favorite storages

Storage templates can be added to a list of favorites. This is used to distinguish the most important templates amongst the many available. See List storages on how to retrieve a list of favorite storages. See Add storage to favorites and Remove storage from favorites on how to manage favorite storages.

8.6. List storages

Returns a list of all accessible storages. The list can be narrowed down by the storage's access type (*public* or *private*), storage type (*normal*, *backup*, *cdrom* or *template*) or favorite status.

Requests

```
GET /1.2/storage
GET /1.2/storage/public
GET /1.2/storage/private
GET /1.2/storage/normal
GET /1.2/storage/backup
GET /1.2/storage/cdrom
GET /1.2/storage/template
GET /1.2/storage/favorite
```

Normal response

```
HTTP/1.0 200 OK
{
  "storages" : {
    "storage" : [
      {
        "access" : "private",
        "license" : 0,
        "size" : 10
        "state" : "online",
        "tier" : hdd",
        "title" : "Operating system disk",
        "type" : "normal",
        "uuid" : "01eff7ad-168e-413e-83b0-054f6a28fa23",
        "zone" : "uk-lon1",
      },
      {
        "access" : "private",
        "license" : 0,
        "size" : 50
        "state" : "online",
        "tier": "maxiops",
        "title" : "Databases",
        "type" : "normal",
        "uuid" : "01f3286c-a5ea-4670-8121-d0b9767d625b",
        "zone" : "fi-hel1"
      }
    ]
  }
}
```

8.7. Get storage details

Returns detailed information about a specific storage resource.

The servers attached to this storage are listed in the *servers* block of the response body.

Request

```
GET /1.2/storage/01d4fcd4-e446-433b-8a9c-551a1284952e
```

Normal response

```
HTTP/1.0 200 OK
{
   "storage" : {
      "access" : "private",
      "backup_rule": "",
      "backups" : {
         "backup" : []
      },
      "license" : 0,
      "servers" : {
         "server" : [
             "00798b85-efdc-41ca-8021-f6ef457b8531"
         ]
      },
      "size" : 10,
      "state" : "online",
      "tier" : "maxiops",
      "title" : "Operating system disk",
      "type" : "normal",
      "uuid" : "01d4fcd4-e446-433b-8a9c-551a1284952e",
      "zone" : "fi-hel1"
   }
}
```

8.8. Create storage

Creates a new storage resource.

Storages are always created to a specific zone. Storages can be attached to servers within the same zone. See List available zones on how to retrieve a list of available zones.

A newly created storage is not attached to any server. See Attach storage to server.

Request

```
POST /1.2/storage
{
  "storage" : {
    "size" : "10",
    "tier" : "maxiops",
    "title" : "My data collection",
    "zone" : "fi-hel1",
    "backup_rule": {
      "interval" : "daily",
      "time" : "0430",
      "retention" : "365"
    }
  }
}
```

Attributes

Attribute	Accepted values	Default value	Required	Description
size	*10-1024*		yes	The size of the storage in gigabytes.
tier	*hdd* / *maxiops*	*hdd*	no	The storage tied to use. See Storage tiers.
title	0-64 characters		yes	A short description.
zone	A valid zone identifier		yes	The zone in which the storage will be created, e.g. *fi-hel1*. See Zones.
backup_rule	A *backup_rule* block		no	The *backup_rule* block defines when the storage device is backed up automatically.

The *backup_rule* block contains the following attributes.

Attribute	Accepted values	Default value	Required	Description
interval	*daily* / *mon* / *tue* / *wed* / *thu* / *fri* / *sat* / *sun*		yes	The weekday when the backup is created. If *daily* is selected, backups are made every day at the same time.
time	*0000-2359*		yes	The time of day when the backup is created.
retention	*1-1095*		yes	The number of days before a backup is automatically deleted. The maximum retention period is three years (1095 days).

If one of the *interval*, *time* or *retention* attributes is specified, others must also be specified. The *backup_rule* block can also be left empty.

Normal response

```
HTTP/1.0 201 Created
{
    "storage" : {
        "access" : "private",
        "backup_rule" [
        "interval" : "daily",
        "hour" : "0430",
        "retention" : "365"
        ],
        "backups" : {
            "backup" : []
        }
        "license" : 0,
        "servers" : {
            "server" : []
        },
        "size" : 10,
        "state" : "online",
        "tier" : "maxiops",
        "title" : "test",
        "type" : "normal",
        "uuid" : "01d4fcd4-e446-433b-8a9c-551a1284952e",
        "zone" : "fi-hel1"
    }
}
```

Error responses

HTTP status	Error code	Description
400 Bad Request	BACKUP_RULE_INVALID	The `backup_rule` block has an invalid value.
400 Bad Request	INTERVAL_INVALID	The attribute `interval` has an invalid value.
400 Bad Request	INTERVAL_MISSING	The required attribute `interval` is missing from the request.
400 Bad Request	HOUR_INVALID	The attribute `hour` has an invalid value.
400 Bad Request	HOUR_MISSING	The required attribute `hour` is missing from the request.
400 Bad Request	RETENTION_INVALID	The attribute `retention` has an invalid value.
400 Bad Request	RETENTION_MISSING	The required attribute `retention` is missing from the request.
400 Bad Request	SIZE_INVALID	The attribute `size` has an invalid value.
400 Bad Request	SIZE_MISSING	The required attribute `size` is missing from the request.
400 Bad Request	TIER_INVALID	The attribute `tier` has an invalid value.
400 Bad Request	TITLE_INVALID	The attribute `title` has an invalid value.
400 Bad Request	TITLE_MISSING	The required attribute `title` is missing from the request.
400 Bad Request	ZONE_INVALID	The attribute `zone` has an invalid value.
400 Bad Request	ZONE_MISSING	The required attribute `zone` is missing from the request.
402 Payment Required	INSUFFICIENT_CREDITS	There are not enough credits to perform the requested action. See Credits.
404 Not Found	ZONE_NOT_FOUND	The zone does not exist.
409 Conflict	STORAGE_RESOURCES_UNAVAILABLE	There are not enough storage resources available in the specified zone and tier to create the requested storage.

8.9. Modify storage

Modifies an existing storage resource. This operation is used to rename or resize the storage.

If the storage is going to be resized, the new size must be greater than the old size. The storage state must be *online* and the storage needs to be detached from any servers. Note that neither the partition table nor the file system on the storage device is resized and such changes have to be made once the storage is attached to a server. A partition table change and a filesystem resize is required to use the available space.

Transferring storage resources between zones is possible using the Clone storage operation.

Request

```
PUT /1.2/storage/011d671f-e803-484d-920a-c25b4bb05c01
{
  "storage" : {
    "size" : "20",
    "title" : "A larger storage"
  }
}
```

Attributes

Attribute	Accepted values	Default value	Required	Description
backup_rule	A *backup_rule* block.		no	A backup rule used to schedule automatic backups on the storage.
size	10-1024		no	The size of the storage in gigabytes.
title	0-64 characters		no	A short description.

The *backup_rule* block contains the following attributes.

Attribute	Accepted values	Default value	Required	Description
interval	*daily* / *mon* / *tue* / *wed* / *thu* / *fri* / *sat* / *sun*		yes	The weekday when the backup is created. If *daily* is selected, backups are made every day at the same time.
time	*0000-2359*		yes	The time of day when the backup is created.
retention	*1-1095*		yes	The number of days before a backup is automatically deleted. The maximum retention period is three years (1095 days).

If one of the *interval*, *time* or *retention* attributes is specified, others must also be specified. The *backup_rule* block can also be left empty.

Normal response

```
HTTP/1.0 202 Accepted
{
   "storage" : {
      "access" : "private",
      "backup_rule" : "",
      "backups" : {
         "backup" []
      },
      "license" : 0,
      "servers" : {
         "server" : []
      },
      "size" : 20,
      "state" : "online",
      "tier" : "maxiops",
      "title" : "A larger storage",
      "type" : "normal",
      "uuid" : "011d671f-e803-484d-920a-c25b4bb05c01",
      "zone" : "uk-lon1"
   }
}
```

Error responses

HTTP status	Error code	Description
400 Bad Request	BACKUP_RULE_INVALID	The *backup_rule* block has an invalid value.
400 Bad Request	INTERVAL_INVALID	The attribute *interval* has an invalid value.
400 Bad Request	INTERVAL_MISSING	The required attribute *interval* is missing from the request.
400 Bad Request	HOUR_INVALID	The attribute *hour* has an invalid value.
400 Bad Request	HOUR_MISSING	The required attribute *hour* is missing from the request.
400 Bad Request	RETENTION_INVALID	The attribute *retention* has an invalid value.
400 Bad Request	RETENTION_MISSING	The required attribute *retention* is missing from the request.
400 Bad Request	SIZE_INVALID	The attribute *size* has an invalid value. If *action* is *clone*, *size* must be greater than the size of the cloned storage.
400 Bad Request	STORAGE_INVALID	The attribute *storage* has an invalid value.
400 Bad Request	TITLE_INVALID	The attribute *title* has an invalid value.
403 Forbidden	STORAGE_FORBIDDEN	The storage exists, but is owned by another account.
404 Not Found	STORAGE_NOT_FOUND	The storage does not exist.
409 Conflict	STORAGE_ATTACHED	The storage is currently attached to a server. To resize a storage, it must first be detached from any servers.
409 Conflict	STORAGE_RESOURCES_UNAVAILABLE	There are not enough storage resources available in the specified zone to resize the storage.
409 Conflict	STORAGE_STATE_ILLEGAL	The storage is in a state in which it cannot be used. See Storage states.
409 Conflict	STORAGE_TYPE_ILLEGAL	The type of the storage to be attached is illegal. Only storages of type *normal* can be resized. See Storage types.

8.10. Attach storage

Attaches a storage as a device to a server.

Attaching normal storages

In order to attach an IDE device, the server state must be *stopped*. SCSI and virtio devices may also be attached while the server state is *started*.

Attaching CD-ROM devices

A storage resource can be attached as a *disk* or a *cdrom*. Only one *cdrom* device may attached on a server at the same time. A CD-ROM device can be empty, i.e. it does not need to have a storage loaded. The storage loaded to the CD-ROM device may be changed by using the Eject CD-ROM and Load CD-ROM operations.

In order to attach a CD-ROM device, the server state must be *stopped*.

Request

```
POST /1.2/server/009d64ef-31d1-4684-a26b-c86c955cbf46/storage/attach
{
  "storage_device" : {
    "type" : "disk",
    "address" : "scsi:0:0",
    "storage" : "00798b85-efdc-41ca-8021-f6ef457b8531"
  }
}
```

Attributes

Attribute	Accepted values	Default value	Required	Description
type	*disk* / *cdrom*	*disk*	no	The type of the attached storage.
address	*ide[01]:[01]* / *scsi:0:[0-7]* / *virtio:[0-7]*	Next available	no	The address where the storage device is attached on the server.
storage	A valid storage UUID		yes if *type* is *disk*	The UUID of the storage to attach.

Normal response

```
HTTP/1.0 200 OK
{
   "server" : {
      "boot_order" : "disk",
      "core_number" : "4",
      "firewall" : "on",
      "hostname" : "debian.example.com",
      "ip_addresses" : {
         "ip_address" : [
            {
               "access" : "private",
               "address" : "10.0.0.0"
            },
            {
               "access" : "public",
               "address" : "x.x.x.x",
               "family" : "IPv4"
            },
            {
               "access" : "public",
               "address" : "xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx",
               "family" : "IPv6"
            }
         ]
      },
      "license" : 0,
      "memory_amount" : "8192",
      "nic_model" : "e1000",
      "password" : "aabbccdd",
      "state" : "started",
      "storage_devices" : {
         "storage_device" : [
            {
               "address" : "scsi:0:0",
               "storage" : "00798b85-efdc-41ca-8021-f6ef457b8531",
               "storage_size" : "10",
               "storage_title" : "Debian from a template",
               "type" : "disk"
            }
         ]
      },
      "timezone" : "UTC",
      "title" : "My Debian server",
      "username" : "root",
      "uuid" : "00c78863-db86-44ea-af70-d6edc4d162bf",
      "video_model" : "cirrus",
      "vnc_password" : "aabbccdd",
      "vnc" : "off",
      "zone" : "fi-hel1"
   }
}
```

Error responses

HTTP status	Error code	Description
400 Bad Request	ADDRESS_INVALID	The attribute *address* has an invalid value.
400 Bad Request	SERVER_INVALID	The server UUID has an invalid value.

HTTP status	Error code	Description
400 Bad Request	STORAGE_INVALID	The attribute *storage* has an invalid value.
400 Bad Request	STORAGE_MISSING	The required attribute *storage* is missing from the request.
400 Bad Request	TYPE_INVALID	The attribute *type* has an invalid value.
403 Forbidden	SERVER_FORBIDDEN	The server exists, but is owned by another account.
403 Forbidden	STORAGE_FORBIDDEN	The storage exists, but is owned by another account.
404 Not Found	SERVER_NOT_FOUND	The server does not exist.
404 Not Found	STORAGE_NOT_FOUND	The storage does not exist.
409 Conflict	CDROM_DEVICE_IN_USE	There is already a CD-ROM device attached on the server. Only one CD-ROM device can be attached per server.
409 Conflict	CDROM_HOTPLUG_UNSUPPORTED	Hotplugging CD-ROM devices is not supported.
409 Conflict	DEVICE_ADDRESS_IN_USE	The device address is already in use.
409 Conflict	IDE_HOTPLUG_UNSUPPORTED	Hotplugging IDE devices is not supported.
409 Conflict	PUBLIC_STORAGE_ATTACH	Attaching a storage of access type *public* is not allowed. See [Storage access types]().
409 Conflict	SERVER_STATE_ILLEGAL	The server is in a state in which it cannot be used. See [Server states]().
409 Conflict	STORAGE_ATTACHED	The storage is already attached to the specified server.
409 Conflict	STORAGE_ATTACHED_AS_CDROM	The storage to be attached as a disk is already attached to some server as a CD-ROM.
409 Conflict	STORAGE_ATTACHED_AS_DISK	The storage to be attached as a CD-ROM is already attached to some server as a disk.
409 Conflict	STORAGE_DEVICE_LIMIT_REACHED	The limit of the number of attached devices has been reached. It is possible to attach four storage devices per server.
409 Conflict	STORAGE_STATE_ILLEGAL	The storage is in a state in which it cannot be used. See [Storage states]().
409 Conflict	STORAGE_TYPE_ILLEGAL	The type of the storage to be attached is illegal. Attaching a cdrom or a template as a disk is not possible. See [Storage types]().
409 Conflict	ZONE_MISMATCH	The storage is located in a different zone than the one the server is created in.

8.11. Detach storage

Detaches a storage resource from a server.

The detached storage is identified by the device address on the attached server.

IDE devices cannot be detached while the server state is *started*.

Request

```
POST /1.2/server/009d64ef-31d1-4684-a26b-c86c955cbf46/storage/detach
{
  "storage_device" : {
    "address" : "scsi:0:0",
  }
}
```

Attributes

Attribute	Accepted values	Default value	Required	Description
address	*ide:[01]:[01] / scsi:0:[0-7] / virtio:[0-7]*		yes	Detach the storage attached to this address.

Normal response

```
HTTP/1.0 200 OK
{
   "server" : {
      "boot_order" : "disk",
      "core_number" : "4",
      "firewall" : "on",
      "hostname" : "debian.example.com",
      "ip_addresses" : {
         "ip_address" : [
            {
               "access" : "private",
               "address" : "10.0.0.0"
            },
            {
               "access" : "public",
               "address" : "x.x.x.x",
               "family" : "IPv4"
            },
            {
               "access" : "public",
               "address" : "xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx",
               "family" : "IPv6"
            }
         ]
      },
      "license" : 0,
      "memory_amount" : "8192",
      "nic_model" : "virtio",
      "password" : "aabbccdd",
      "state" : "started",
      "storage_devices" : {
      "storage_device" : [ ]
      },
      "timezone" : "UTC",
      "title" : "My Debian server",
      "username" : "root",
      "uuid" : "00c78863-db86-44ea-af70-d6edc4d162bf",
      "video_model" : "cirrus",
      "vnc" : "off",
      "vnc_password" : "aabbccdd",
      "zone" : "fi-hel1"
   }
}
```

Error responses

HTTP status	Error code	Description
400 Bad Request	ADDRESS_INVALID	The attribute *address* has an invalid value.
400 Bad Request	ADDRESS_MISSING	The required attribute *address* is missing from the request.
400 Bad Request	SERVER_INVALID	The server UUID has an invalid value.
403 Forbidden	SERVER_FORBIDDEN	The server exists, but is owned by another account.
404 Not Found	SERVER_NOT_FOUND	The server does not exist.
409 Conflict	CDROM_HOTPLUG_UNSUPPORTED	Hotplugging CD-ROM devices is not supported.
409 Conflict	DEVICE_ADDRESS_NOT_IN_USE	There was no storage device attached to the specified device address.
409 Conflict	STORAGE_STATE_ILLEGAL	The storage is in a state in which it cannot be used. See Storage states.
409 Conflict	SERVER_STATE_ILLEGAL	The server is in a state in which it cannot be used. See Server states.
409 Conflict	IDE_HOTPLUG_UNSUPPORTED	Hotplugging IDE devices is not supported.
511 Operation Failed	HOTPLUG_FAILED	The hotplug operation failed.

8.12. Load CD-ROM

Loads a storage as a CD-ROM in the CD-ROM device of a server.

This operation requires that a CD-ROM device is attached to the server. A CD-ROM device can be attached using the Attach storage operation.

Any storage of type *normal*, *cdrom or backup* can be loaded as a CD-ROM.

Request

```
POST /1.2/server/00798b85-efdc-41ca-8021-f6ef457b8531/cdrom/load
{
  "storage_device" : {
    "storage" : "01000000-0000-4000-8000-000060010101"
  }
}
```

Attributes

Attribute	Accepted values	Default value	Required	Description
storage	UUID of a storage of type *normal*, *cdrom* or *backup*.		yes	The UUID of the storage to be loaded in the CD-ROM device.

Normal response

```
HTTP/1.0 200 OK
{
   "server" : {
      "boot_order" : "cdrom,disk",
      "core_number" : "0",
      "firewall" : "on",
      "hostname" : "test.example.com",
      "ip_addresses" : {
         "ip_address" : [
            {
               "access" : "private",
               "address" : "10.0.0.0"
            },
            {
               "access" : "public",
               "address" : "x.x.x.x",
               "family" : "IPv4"
            },
            {
               "access" : "public",
               "address" : "xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx",
               "family" : "IPv6"
            }
         ]
      },
      "license": 0,
      "memory_amount" : "512",
      "nic_model : "virtio",
      "state" : "started",
      "storage_devices" : {
         "storage_device" : [
            {
               "type" : "cdrom",
               "address" : "ide:0:1",
               "storage" : "01000000-0000-4000-8000-000060010101"
            },
            {
               "type" : "disk",
               "address" : "ide:0:0",
               "storage" : "01d4fcd4-e446-433b-8a9c-551a1284952e"
            }
         ]
      },
      "timezone" : "UTC",
      "title" : "Example server"
      "uuid" : "00798b85-efdc-41ca-8021-f6ef457b8531",
      "vnc_password" : "aabbccdd",
      "vnc" : "off",
      "zone" : "fi-hel1"
   }
}
```

Error responses

HTTP status	Error code	Description
400 Bad Request	STORAGE_INVALID	The attribute $storage$ has an invalid value.
400 Bad Request	STORAGE_MISSING	The required attribute $storage$ is missing from the request.
400 Bad Request	SERVER_INVALID	The server UUID has an invalid value.
403 Forbidden	SERVER_FORBIDDEN	The server exists, but it is owned by another account.
403 Forbidden	STORAGE_FORBIDDEN	The server exists, but is owned by another account.
404 Not Found	SERVER_NOT_FOUND	The server does not exist.
404 Not Found	STORAGE_NOT_FOUND	The storage does not exist.
409 Conflict	NO_CDROM_DEVICE	There is no CD-ROM device attached to the server.
409 Conflict	SERVER_STATE_ILLEGAL	The server is in a state in which it cannot be used. See Server states.
409 Conflict	STORAGE_STATE_ILLEGAL	The storage is in a state in which it cannot be used. See Storage states.
409 Conflict	STORAGE_TYPE_ILLEGAL	The type of the storage to be attached is illegal. Only storages of type $normal$ and $cdrom$ can be loaded as CD-ROMs.
409 Conflict	STORAGE_ATTACHED	The specified storage is already loaded as a CD-ROM in the specified server.
409 Conflict	STORAGE_ATTACHED_AS_DISK	The storage to be loaded is already attached to some server as a disk.
409 Conflict	ZONE_MISMATCH	The storage is located in a different zone than the server.

8.13. Eject CD-ROM

Ejects the storage from the CD-ROM device of a server.

Request

```
POST /1.2/server/00798b85-efdc-41ca-8021-f6ef457b8531/cdrom/eject
```

Normal response

```
HTTP/1.0 200 OK
{
   "server" : {
      "boot_order" : "cdrom,disk",
      "core_number" : "0",
      "firewall" : "on",
      "hostname" : "test.example.com",
      "ip_addresses" : {
         "ip_address" : [
            {
               "access" : "private",
               "address" : "10.0.0.0"
            },
            {
               "access" : "public",
               "address" : "x.x.x.x",
               "family" : "IPv4"
            },
            {
               "access" : "public",
               "address" : "xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx",
               "family" : "IPv6"
            }
         ]
      },
      "license": 0,
      "memory_amount" : "512",
      "nic_model : "virtio",
      "state" : "started",
      "storage_devices" : {
         "storage_device" : [
            {
               "type" : "cdrom",
               "address" : "ide:0:1",
               "storage" : ""
            },
            {
               "type" : "disk",
               "address" : "ide:0:0",
               "storage" : "01d4fcd4-e446-433b-8a9c-551a1284952e"
            }
         ]
      },
      "timezone" : "UTC",
      "title" : "Example server"
      "uuid" : "00798b85-efdc-41ca-8021-f6ef457b8531",
      "vnc_password" : "aabbccdd",
      "vnc" : "off",
      "zone" : "fi-hel1"
   }
}
```

Error responses

HTTP status	Error code	Description
400 Bad Request	SERVER_INVALID	The server UUID has an invalid value.
403 Forbidden	SERVER_FORBIDDEN	The server exists, but is owned by another account.
404 Not Found	SERVER_NOT_FOUND	The server does not exist.
409 Conflict	NO_CDROM_DEVICE	There is no CD-ROM device attached to the server.
409 Conflict	SERVER_STATE_ILLEGAL	The server is in a state in which it cannot be used. See Server states.
409 Conflict	STORAGE_STATE_ILLEGAL	The storage is in a state in which it cannot be used. See Storage states.
511 Operation Failed	CDROM_EJECT_FAILED	Could not eject the CD-ROM. It may still be mounted on the server.

8.14. Clone storage

Creates an exact copy of an existing storage resource.

The speed of the clone operation depends on the size of the storage and whether the source and target storages are in the same zone. A clone operation between different zones takes a considerably longer time than a clone operation within the same zone.

This operation is asynchronous. The state of the cloned storage is `maintenance` and changes to `online` after the clone process is complete. The status of the operation can be monitored by polling the storage with the Get storage details operation.

Request

```
POST /1.2/storage/01eff7ad-168e-413e-83b0-054f6a28fa23/clone
{
  "storage" : {
    "zone" : "fi-hel1",
    "tier" : "maxiops",
    "title" : "Clone of operating system disk"
  }
}
```

Attributes

Attribute	Accepted values	Default value	Required	Description
zone	Valid zone id		yes	The zone in which the storage will be created, e.g. fi-hel1. See Zones.
tier	*hdd* / *maxiops*	*hdd*	no	The storage tier to use. See Storage tiers.
title	0-64 characters		yes	A short description.

Normal response

```
HTTP/1.0 201 Created
{
  "storage" : {
    "access" : "private",
    "license" : 0,
    "servers" : {
      "server" : [
      ]
    },
    "size" : "10",
    "state" : "maintenance",
    "tier" : "maxiops",
    "title" : "Clone of operating system disk",
    "type" : "normal",
    "uuid" : "01f3286c-a5ea-4670-8121-d0b9767d625b",
    "zone" : "fi-hel1"
  }
}
```

Error responses

HTTP status	Error code	Description
400 Bad Request	STORAGE_INVALID	The storage UUID has an invalid value.
400 Bad Request	TIER_INVALID	The attribute $tier$ has an invalid value.
400 Bad Request	TITLE_INVALID	The attribute $title$ has an invalid value.
400 Bad Request	TITLE_INVALID	The required attribute $title$ is missing from the request.
400 Bad Request	ZONE_INVALID	The attribute $zone$ has an invalid value.
400 Bad Request	ZONE_MISSING	The required attribute $zone$ is missing from the request.
402 Payment Required	INSUFFICIENT_CREDITS	There are not enough credits to perform the requested action. See Credits.
403 Forbidden	STORAGE_FORBIDDEN	The storage exists, but is owned by another account.
404 Not Found	STORAGE_NOT_FOUND	The storage does not exist.
404 Not Found	ZONE_NOT_FOUND	The zone does not exist.
409 Conflict	STORAGE_RESOURCES_UNAVAILABLE	There were not enough storage resources available in the specified zone to create the storage.
409 Conflict	STORAGE_STATE_ILLEGAL	The storage is in a state in which it cannot be used. See Storage states.
409 Conflict	STORAGE_TYPE_ILLEGAL	The type of the storage to be attached is illegal. See Storage types.

8.15. Create backup

Creates a point-in-time backup of a storage resource.

This operation is asynchronous. The state of the storage changes to *backuping* and changes back to *online* after the backup process is complete. The status of the operation can be monitored by polling the storage with the Get storage details operation.

Request

```
POST /1.2/storage/01eff7ad-168e-413e-83b0-054f6a28fa23/backup

{
  "storage" : {
    "title" : "Manually created backup"
  }
}
```

Attributes

Attribute	Accepted values	Default value	Required	Description
title	0-64 characters		yes	A short description.

Normal response

```
HTTP/1.0 201 Created

{
  "storage" : {
    "access" : "private",
    "created" : "2015-03-27T10:02:05Z",
    "license" : 0,
    "origin" : "01eff7ad-168e-413e-83b0-054f6a28fa23",
    "progress": "0",
    "servers" : {
      "server" : [
      ]

    },
    "size" : "10",
    "state" : "maintenance",

    "title" : "Manually created backup",
    "type" : "normal",

    "uuid" : "01f3286c-a5ea-4670-8121-d0b9767d625b",
    "zone" : "fi-hel1"
  }
}
```

Error responses

HTTP status	Error code	Description
400 Bad Request	STORAGE_INVALID	The storage UUID has an invalid value.
400 Bad Request	TITLE_INVALID	The attribute *title* has an invalid value.
400 Bad Request	TITLE_MISSING	The required attribute *title* is missing from the request.
402 Payment Required	INSUFFICIENT_CREDITS	There are not enough credits to perform the requested action. See Credits.
403 Forbidden	STORAGE_FORBIDDEN	The storage exists, but is owned by another account.
404 Not Found	STORAGE_NOT_FOUND	The storage does not exist.
409 Conflict	STORAGE_RESOURCES_UNAVAILABLE	There were not enough storage resources available in the specified zone to create the storage.
409 Conflict	STORAGE_STATE_ILLEGAL	The storage is in a state in which it cannot be used. See Storage states.
409 Conflict	STORAGE_TYPE_ILLEGAL	The type of the storage to be attached is illegal. See Storage types.

8.16. Restore backup

Restores the origin storage with data from the specified backup storage.

If the origin storage is attached to a server, the server must first be stopped.

Request

```
POST /1.2/storage/01f3286c-a5ea-4670-8121-d0b9767d625b/restore
```

Response

```
HTTP/1.0 204 No Content
```

Error responses

HTTP status	Error code	Description
400 Bad Request	STORAGE_INVALID	The storage UUID has an invalid value.
400 Bad Request	TITLE_INVALID	The attribute *title* has an invalid value.
400 Bad Request	TITLE_INVALID	The required attribute *title* is missing from the request.
400 Bad Request	ZONE_INVALID	The attribute *zone* has an invalid value.
400 Bad Request	ZONE_MISSING	The required attribute *zone* is missing from the request.
402 Payment Required	INSUFFICIENT_CREDITS	There are not enough credits to perform the requested action. See Credits.
403 Forbidden	STORAGE_FORBIDDEN	The storage exists, but is owned by another account.
404 Not Found	STORAGE_NOT_FOUND	The storage does not exist.
404 Not Found	ZONE_NOT_FOUND	The zone does not exist.
409 Conflict	SERVER_STATE_ILLEGAL	The origin storage is attached to a started server. The server must be stopped in order to revert the storage.
409 Conflict	STORAGE_RESOURCES_UNAVAILABLE	There were not enough storage resources available in the specified zone to create the storage.
409 Conflict	STORAGE_STATE_ILLEGAL	The storage is in a state in which it cannot be used. See Storage states.
409 Conflict	STORAGE_TYPE_ILLEGAL	The type of the storage to be attached is illegal. See Storage types.

8.17. Add storage to favorites

Adds a storage to the list of favorite storages. To list favorite storages, see List storages. This operations succeeds even if the storage is already on the list of favorites.

Request

```
POST /1.2/storage/01f3286c-a5ea-4670-8121-d0b9767d625b/favorite
```

Response

```
HTTP/1.0 204 No Content
```

Error responses

HTTP status	Error code	Description
400 Bad Request	STORAGE_INVALID	The storage UUID has an invalid value.
403 Forbidden	STORAGE_FORBIDDEN	The storage exists, but is owned by another account.
404 Not Found	STORAGE_NOT_FOUND	The storage does not exist.

8.18. Remove storage from favorites

Adds a storage to the list of favorite storages. To list favorite storages, see List storages. This operations succeeds even if the storage is already on the list of favorites.

Request

```
DELETE /1.2/storage/01f3286c-a5ea-4670-8121-d0b9767d625b/favorite
```

Response

```
HTTP/1.0 204 No Content
```

Error responses

HTTP status	Error code	Description
400 Bad Request	STORAGE_INVALID	The storage UUID has an invalid value.
403 Forbidden	STORAGE_FORBIDDEN	The storage exists, but is owned by another account.
404 Not Found	STORAGE_NOT_FOUND	The storage does not exist.

8.19. Delete storage

Deleted an existing storage resource.

The state of the storage must be *online*. The storage must not be attached to any server.

Backups of the deleted storage resource are not deleted and can be used to restore the deleted storage resource.

Request

```
DELETE /1.2/storage/01d4fcd4-e446-433b-8a9c-551a1284952e
```

Normal response

```
HTTP/1.0 204 No Content
```

Error responses

HTTP status	Error code	Description
400 Bad Request	STORAGE_INVALID	The attribute *storage* has an invalid value.
403 Forbidden	STORAGE_FORBIDDEN	The storage exists, but is owned by another account.
404 Not Found	STORAGE_NOT_FOUND	The storage does not exist.
409 Conflict	STORAGE_ATTACHED	The storage is currently attached to a server.
409 Conflict	STORAGE_STATE_ILLEGAL	The storage is in a state in which it cannot be used.

9. IP ADDRESSES

The UpCloud network has public and private IP addresses.

Public IP addresses are used to connect to the server from the Internet. Private addresses are only visible to servers under the same UpCloud account. A server has a minimum of one and a maximum of five public IPv4 and IPv6 addresses. Every server can have one private IPv4 address only. The UpCloud API assigns the IP addresses automatically. IP addresses cannot be reserved for later use.

The operating system running on the server will see a separate network interface per IP address. On a typical server, there are three network interfaces. The first network interface is assigned to the public network and the second to a private network. If the server has an IPv6 address configured it will use the third network interface. DHCP service is provided on all IPv4 network interfaces for automatic configuration of the correct IP address. IPv6 network interfaces should use stateless address autoconfiguration (SLAAC) with IPv6 privacy extensions disabled. On IPv6-only server DNS parameters can be obtained with stateless DHCPv6.

9.1. List IP addresses

Returns a list of all IP addresses assigned to servers on the current user account.

Request

```
GET /1.2/ip_address
```

Normal response

```
HTTP/1.0 200 OK
{
  "ip_addresses" : {
    "ip_address" : [
        {
          "access" : "private",
          "address" : "10.0.0.0",
          "family" : "IPv4",
          "ptr_record" : "",
          "server" : "0053cd80-5945-4105-9081-11192806a8f7"
        },
        {
          "access" : "private",
          "address" : "10.0.0.1"
          "family" : "IPv4",
          "ptr_record" : "",
          "server" : "006b6701-55d2-4374-ac40-56cc1501037f"
        },
        {
          "access" : "public",
          "address" : "x.x.x.x,
          "family" : "IPv4",
          "ptr_record": "x.x.x.x.zone.host.upcloud.com",
          "server" : "0053cd80-5945-4105-9081-11192806a8f7"
        },
        {
          "access" : "public",
          "address" : "xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx",
          "family" : "IPv6",
          "ptr_record" : "xxxx-xxxx-xxxx-xxxx.v6.zone.host.upcloud.com",
          "server" : "006b6701-55d2-4374-ac40-56cc1501037f"
        }
    ]
  }
}
```

9.2. Get IP address details

Returns detailed information about a specific IP address.

Request

```
GET /1.2/ip_address/0.0.0.0
```

Normal response

```
HTTP/1.0 200 OK
{
  "ip_address" : {
    "access" : "public",
    "family" : "IPv4",
    "server" : "009d64ef-31d1-4684-a26b-c86c955cbf46",
    "address" : "0.0.0.0"
  }
}
```

9.3. Assign IP address

Assigns a new IP address to a server.

Only public IP addresses can be added. There is always exactly one private IP address per server. There is a maximum of five public IP addresses per server.

The server state must be *stopped*.

Request

```
POST /1.2/ip_address
{
  "ip_address" : {
    "family" : "IPv4",
    "server" : "009d64ef-31d1-4684-a26b-c86c955cbf46"
  }
}
```

Attributes

Attribute	Accepted values	Default value	Required	Description
access	*private / public*	public	yes	Is address for private or public network.
family	*IPv4 / IPv6*	IPv4	no	The address family of new IP address.

Normal response

```
HTTP/1.0 201 Created
{
  "ip_address" : {
    "access" : "public",
    "family" : "IPv4",
    "server" : "009d64ef-31d1-4684-a26b-c86c955cbf46",
    "address" : "0.0.0.0",
  }
}
```

Error responses

HTTP status	Error code	Description
400 Bad Request	SERVER_INVALID	The attribute *server* has an invalid value.
402 Payment Required	INSUFFICIENT_CREDITS	There are not enough credits to perform the requested action. See Credits.
403 Forbidden	SERVER_FORBIDDEN	The server exists, but is owned by another account.
404 Not Found	SERVER_NOT_FOUND	The server does not exist.
409 Conflict	IP_ADDRESS_LIMIT_REACHED	The maximum number of IP addresses are already attached to the server.
409 Conflict	IP_ADDRESS_RESOURCES_UNAVAILABLE	There are not enough IP addresses available in the specified zone.
409 Conflict	SERVER_STATE_ILLEGAL	The server is in a state in which it cannot be used. See Server states.

9.4. Modify IP address

Modifies the reverse DNS PTR record corresponding to an IP address. The PTR record can only be set to public IP address.

Request

```
PUT /1.2/ip_address/0.0.0.0
{
  "ip_address" : {
    "ptr_record" : "hostname.example.com"
  }
}
```

Attributes

Attribute	Accepted values	Default value	Required	Description
ptr_record	DNS PTR record value		yes	A fully qualified domain name.

Normal response

```
HTTP/1.0 202 Accepted
{
  "ip_address" : {
    "access" : "public",
    "address" : "0.0.0.0",
         "family" : "IPv4",
    "ptr_record" : hostname.example.com",
    "server" : "009d64ef-31d1-4684-a26b-c86c95c5bf46"
  }
}
```

Error responses

HTTP status	Error code	Description
400 Bad Request	IP_ADDRESS_INVALID	The attribute *ip_address* has an invalid value.
400 Bad Request	PTR_RECORD_INVALID	The attribute *ptr_record* has an invalid value.
403 Forbidden	IP_ADDRESS_FORBIDDEN	The IP address exists, but belong to another user.
404 Not found	IP_ADDRESS_NOT_FOUND	The IP address does not exist.
409 Conflict	PTR_RECORD_NOT_SUPPORTED	A PTR record was tried to set on a private IP address.

9.5. Release IP

Removes an IP address from a server.

Request

```
DELETE /1.2/0.0.0.0
```

Normal response

```
HTTP/1.0 204 No Content
```

Error responses

HTTP status	Error code	Description
400 Bad Request	IP_ADDRESS_INVALID	The attribute *ip_address* has an invalid value.
403 Forbidden	IP_ADDRESS_FORBIDDEN	The IP address exists, but is assigned to another account.
404 Not Found	IP_ADDRESS_NOT_FOUND	The IP address does not exist.
409 Conflict	SERVER_STATE_ILLEGAL	The server is in a state in which it cannot be used. See Server states.
409 Conflict	CANNOT_DELETE_PRIVATE_ADDRESS	The IP address on the private network cannot be deleted.

10. FIREWALL

Each server has its own firewall rules. The firewall is enabled on all network interfaces.

The firewall filters packets by:

- Traffic direction (in, out)
- IP family (IPv4, IPv6)
- Network protocol (TCP, UDP, ICMP)
- Source and/or destination IP addresses
- For TCP and UDP protocols, the source and/or destination ports
- For the ICMP protocol, the packet type

Network traffic is checked against the firewall rules in numerical order starting from the first rule. It is important to add the firewall rules in the correct order to make sure the firewall functions properly. Place the most specific rules first and the more generic rules later.

The firewall takes three different actions based on the applied rules. *Accept* lets the packet pass the firewall. *Reject* blocks the packet and sends an ICMP error message to the source IP address on the packet. *Drop* blocks the traffic.

10.1. List firewall rules

Returns a list of firewall rules for a specific server.

Request

```
GET /1.2/server/00798b85-efdc-41ca-8021-f6ef457b8531/firewall_rule
```

Normal response

```
HTTP/1.0 200 OK
{
  {
    "firewall_rule" : {
      "action" : "accept",
      "destination_address_end" : "",
      "destination_address_start" : "",
      "destination_port_end" : "80",
      "destination_port_start" : "80",
      "direction" : "in",
      "family" : "IPv4",
      "icmp_type" : "",
      "position" : "1",
      "protocol" : "",
      "source_address_end" : "",
      "source_address_start" : "",
      "source_port_end" : "",
      "source_port_start" : ""
    },
    "firewall_rule" : {
      "action" : "accept",
      "destination_address_end" : "",
      "destination_address_start" : "",
      "destination_port_end" : "22",
      "destination_port_start" : "22",
      "direction" : "in",
      "family" : "IPv4",
      "icmp_type" : "",
      "position" : "2",
      "protocol" : "tcp",
      "source_address_end" : "192.168.1.255",
      "source_address_start" : "192.168.1.1",
      "source_port_end" : "",
      "source_port_start" : ""
    },
    "firewall_rule" : {
      "action" : "accept",
      "destination_address_end" : "",
      "destination_address_start" : "",
      "destination_port_end" : "22",
      "destination_port_start" : "22",
      "direction" : "in",
      "family" : "IPv6",
      "icmp_type" : "",
      "position" : "3",
      "protocol" : "tcp",
      "source_address_end" : "2a04:3540:1000:aaaa:bbbb:cccc:d001",
      "source_address_start" : "2a04:3540:1000:aaaa:bbbb:cccc:d001",
      "source_port_end" : "",
      "source_port_start" : ""
    },
```

```
        "firewall_rule" : {
          "action" : "accept",
          "destination_address_end" : "",
          "destination_address_start" : "",
          "destination_port_end" : "",
          "destination_port_start" : "",
          "direction" : "in",
          "family" : "IPv4",
          "icmp_type" : "8",
          "position" : "4",
          "protocol" : "icmp",
          "source_address_end" : "",
          "source_address_start" : "",
          "source_port_end" : "",
          "source_port_start" : "",
        },
        "firewall_rule" : {
          "action" : "drop",
          "destination_address_end" : "",
          "destination_address_start" : "",
          "destination_port_end" : "",
          "destination_port_start" : "",
          "direction" : "in",
          "family" : "",
          "icmp_type" : "",
          "position" : "5",
          "protocol" : "",
          "source_address_end" : "",
          "source_address_start" : "",
          "source_port_end" : "",
          "source_port_start" : ""
        }
      }
```

10.2. Get firewall rule details

Returns detailed information about a specific firewall rule.

Request

```
GET /1.2/server/00798b85-efdc-41ca-8021-f6ef457b8531/firewall_rule/1
```

The last number denotes the index of the firewall rule in the server's firewall rule list. The index of the first rule is 1.

Normal response

```
HTTP/1.0 200 OK
{
    "firewall_rule" : {
      "action" : "accept",
      "destination_address_end" : "",
      "destination_address_start" : "",
      "destination_port_end" : "80",
      "destination_port_start" : "80",
      "direction" : "in",
      "family" : "IPv4",
      "icmp_type" : "",
      "position" : "1",
      "protocol" : "",
      "source_address_end" : "",
      "source_address_start" : "",
      "source_port_end" : "",
      "source_port_start" : ""
    }
}
```

10.3. Create firewall rule

Creates a new firewall rule.

If used, IP address and port ranges must have both start and end values specified. These can be the same value if only one IP address or port number is specified. Source and destination port numbers can only be set if the protocol is TCP or UDP. The ICMP type may only be set if the protocol is ICMP.

The maximum number of firewall rules per server is 1000.

The order of the firewall rules is set with a the *position* attribute. The position of the first rule is always 1. Firewall rule positions are always successive numbers. The rule positions will be adjusted accordingly when rules are added or deleted. The last rule is a special case and corresponds to Default Rule set through Control Panel. It should contain only direction and action attributes in addition of position.

Example request

```
POST /1.2/server/00798b85-efdc-41ca-8021-f6ef457b8531/firewall_rule
{
  "firewall_rule" : {
    "position" : "1",
    "direction" : "in",
    "family" : "IPv4",
    "protocol" : "tcp",
    "source_address_start" : "192.168.1.1",
    "source_address_end" : "192.168.1.255",
    "source_port_end" : "",
    "source_port_start" : "",
    "destination_address_start" : "",
    "destination_address_end" : ""
    "destination_port_start" : "22",
    "destination_port_end" : "22",
    "icmp_type" : "",
    "action" : "accept"
  }
}
```

Attributes

Attribute	Accepted values	Required	Description
direction	in / out	yes	The direction of network traffic this rule will be applied to.
action	accept / reject / drop	yes	Action to take if the rule conditions are met.
position	1-1000	no	Add the firewall rule to this position in the server's firewall list.
family	IPv4 / IPv6	yes if `protocol` is set	The address family of new firewall rule
protocol	tcp / udp / icmp	no	The protocol this rule will be applied to.
icmp_type	0-255	no	The ICMP type.
destination_address_start	Valid IP address	yes if `destination_address_end` is set	The destination address range starts from this address.
destination_address_end	Valid IP address	yes if `destination_address_start` Is set	The destination address range ends to this address.
destination_port_start	1-65535	yes if `destination_port_end` is set	The destination port range starts from this port number.
destination_port_end	1-65535	yes if `destination_port_start` is set	The destination port range ends to this port number.
source_address_start	Valid IP address	yes if `source_address_end` is set	The source address range starts from this address.
source_address_end	Valid IP address	yes if `source_address_start` is set	The source address range ends to this address.
source_port_start	1-65535	yes if `source_port_end` is set	The source port range starts from this port number.
source_port_end	1-65535	yes if `source_port_start` is set	The source port range ends to this port number.

Note: No default value is set to attributes.

Normal response

```
HTTP/1.0 201 Created
{
  "firewall_rule" : {
      "action" : "accept",
      "destination_address_end" : "",
      "destination_address_start" : "",
      "destination_port_end" : "80",
      "destination_port_start" : "80",
      "direction" : "in",
      "family" : "IPv4",
      "icmp_type" : "",
      "position" : "1",
      "protocol" : "",
      "source_address_end" : "",
      "source_address_start" : "",
      "source_port_end" : "",
      "source_port_start" : ""
    }
}
```

Error responses

HTTP status	Error code	Description
400 Bad Request	ACTION_INVALID	The attribute `action` has an invalid value.
400 Bad Request	ACTION_MISSING	The required attribute `action` is missing from the request.
400 Bad Request	ICMP_TYPE_INVALID	The attribute `icmp_type` has an invalid value.
400 Bad Request	DESTINATION_ADDRESS_ORDER_ILLEGAL	The destination end address is smaller than the destination start address.
400 Bad Request	DESTINATION_ADDRESS_START_INVALID	The attribute `destination_address_start` has an invalid value.
400 Bad Request	DESTINATION_ADDRESS_END_INVALID	The attribute `destination_address_end` has an invalid value.
400 Bad Request	DESTINATION_PORT_ORDER_ILLEGAL	The destination end port is smaller than the destination start port.
400 Bad Request	DESTINATION_PORT_START_INVALID	The attribute `destination_port_start` has an invalid value.
400 Bad Request	DESTINATION_PORT_END_INVALID	The attribute `destination_port_end` has an invalid value.
400 Bad Request	ICMP_TYPE_PROTOCOL_MISMATCH	The `icmp_type` attribute was specified, but `protocol` was not `icmp`.
400 Bad Request	PORT_PROTOCOL_MISMATCH	Port numbers were specified but `protocol` was not `tcp` or `udp`.
400 Bad Request	POSITION_INVALID	The attribute `position` has an invalid value.
400 Bad Request	PROTOCOL_INVALID	The attribute `protocol` has an invalid value.
400 Bad Request	SERVER_INVALID	The server UUID was invalid.
400 Bad Request	SOURCE_ADDRESS_ORDER_ILLEGAL	The source end address is smaller than the source start address.
400 Bad Request	SOURCE_ADDRESS_START_INVALID	The attribute `source_address_start` has an invalid value.
400 Bad Request	SOURCE_ADDRESS_END_INVALID	The attribute `source_address_end` has an invalid value.
400 Bad Request	SOURCE_PORT_ORDER_ILLEGAL	The source end port is smaller than the source start port.
400 Bad Request	SOURCE_PORT_START_INVALID	The attribute `source_port_start` has an invalid value.
400 Bad Request	SOURCE_PORT_END_INVALID	The attribute `source_port_end` has an invalid value.
403 Forbidden	SERVER_FORBIDDEN	The server exists, but is owned by another account.

Error responses

HTTP status	Error code	Description
400 Bad Request	POSITION_INVALID	The attribute *position* has an invalid value.
400 Bad Request	SERVER_INVALID	The server UUID has an invalid value.
403 Forbidden	SERVER_FORBIDDEN	The server exists, but is owned by another account.
404 Not Found	SERVER_NOT_FOUND	The server does not exist.
404 Not Found	FIREWALL_RULE_NOT_FOUND	The firewall rule does not exist.
409 Conflict	SERVER_STATE_ILLEGAL	The server is in a state in which it cannot be used. See Server states.