Compute API

In a hybrid environment, the underlying implementation might not control the IP address of a server. Instead, the access IP address might be part of the dedicated hardware; for example, a router/NAT device. In this case, you cannot use the addresses that the implementation provides to access the server from outside the local LAN. Instead, the API might assign a separate access address at creation time to provide access to the server. This address might not be directly bound to a network interface on the server and might not necessarily appear when you query the server addresses. However, clients should use an access address to access the server directly.

Servers (servers)¶

Lists, creates, shows details for, updates, and deletes servers.

Passwords

When you create a server, you can specify a password through the optional adminPass attribute. The password must meet the complexity requirements set by your OpenStack Compute provider. The server might enter an `ERROR` state if the complexity requirements are not met. In this case, a client might issue a change password action to reset the server password.

If you do not specify a password, the API generates and assigns a random password that it returns in the response object. This password meets the security requirements set by the compute provider. For security reasons, subsequent GET calls do not require this password.

Server metadata

You can specify custom server metadata at server launch time. The maximum size for each metadata key-value pair is 255 bytes. The compute provider determines the maximum number of key-value pairs for each server. You can query this value through the `maxServerMeta` absolute limit.

Server networks

You can specify one or more networks to which the server connects at launch time. Users can also specify a specific port on the network or the fixed IP address to assign to the server interface.

> Note
>
> You can use both IPv4 and IPv6 addresses as access addresses, and you can assign both addresses simultaneously. You can update access addresses after you create a server.

Server personality

> Note
>
> The use of personality files is deprecated starting with the 2.57 microversion. Use `metadata` and `user_data` to customize a server instance.

To customize the personality of a server instance, you can inject data into its file system. For example, you might insert ssh keys, set configuration files, or store data that you want to retrieve from inside the instance. This customization method provides minimal launch-time personalization. If you require significant customization, create a custom image.

Follow these guidelines when you inject files:

- The maximum size of the file path data is 255 bytes.
- Encode the file contents as a Base64 string. The compute provider determines the maximum size of the file contents. The image that you use to create the server determines this value.

  > Note
  >
  > The maximum limit refers to the number of bytes in the decoded data and not to the number of characters in the encoded data.

- The `maxPersonality` absolute limit defines the maximum number of file path and content pairs that you can supply. The compute provider determines this value.
- The `maxPersonalitySize` absolute limit is a byte limit that applies to all images in the deployment. Providers can set additional per-image personality limits.

The file injection might not occur until after the server builds and boots.

After file injection, only system administrators can access personality files. For example, on Linux, all files have root as the owner and the root group as the group owner, and allow only user and group read access (`chmod 440`).

Server access addresses

In a hybrid environment, the underlying implementation might not control the IP address of a server. Instead, the access IP address might be part of the dedicated hardware; for example, a router/NAT device. In this case, you cannot use the addresses that the implementation provides to access the server from outside the local LAN. Instead, the API might assign a separate access address at creation time to provide access to the server. This address might not be directly bound to a network interface on the server and

might not necessarily appear when you query the server addresses. However, clients should use an access address to access the server directly.

List Servers GET /servers

Lists IDs, names, and links for all servers.

Servers contain a status attribute that indicates the current server state. You can filter on the server status when you complete a list servers request. The server status is returned in the response body. The possible server status values are:

- `ACTIVE`. The server is active.
- `BUILD`. The server has not finished the original build process.
- `DELETED`. The server is permanently deleted.
- `ERROR`. The server is in error.
- `HARD_REBOOT`. The server is hard rebooting. This is equivalent to pulling the power plug on a physical server, plugging it back in, and rebooting it.
- `MIGRATING`. The server is being migrated to a new host.
- `PASSWORD`. The password is being reset on the server.
- `PAUSED`. In a paused state, the state of the server is stored in RAM. A paused server continues to run in frozen state.
- `REBOOT`. The server is in a soft reboot state. A reboot command was passed to the operating system.
- `REBUILD`. The server is currently being rebuilt from an image.
- `RESCUE`. The server is in rescue mode. A rescue image is running with the original server image attached.
- `RESIZE`. Server is performing the differential copy of data that changed during its initial copy. Server is down for this stage.
- `REVERT_RESIZE`. The resize or migration of a server failed for some reason. The destination server is being cleaned up and the original source server is restarting.
- `SHELVED`: The server is in shelved state. Depending on the shelve offload time, the server will be automatically shelved offloaded.
- `SHELVED_OFFLOADED`: The shelved server is offloaded (removed from the compute host) and it needs unshelved action to be used again.
- `SHUTOFF`. The server is powered off and the disk image still persists.
- `SOFT_DELETED`. The server is marked as deleted but the disk images are still available to restore.
- `SUSPENDED`. The server is suspended, either by request or necessity. This status appears for only the XenServer/XCP, KVM, and ESXi hypervisors. Administrative users can suspend an instance if it is infrequently used or to perform system maintenance. When you suspend an instance, its VM state is stored on disk, all memory is written to disk, and the virtual machine is stopped. Suspending an instance is similar to placing a device in hibernation; memory and vCPUs become available to create other instances.
- `UNKNOWN`. The state of the server is unknown. Contact your cloud provider.
- `VERIFY_RESIZE`. System is awaiting confirmation that the server is operational after a move or resize.

There is whitelist for valid filter keys. Any filter key other than from whitelist will be silently ignored.

- For non-admin users, whitelist is different from admin users whitelist. Valid whitelist for non-admin users includes
  - `all_tenants`
  - `changes-since`
  - `flavor`
  - `image`
  - `ip`
  - `ip6` (New in version 2.5)
  - `name`
  - `not-tags` (New in version 2.26)
  - `not-tags-any` (New in version 2.26)
  - `reservation_id`
  - `status`
  - `tags` (New in version 2.26)
  - `tags-any` (New in version 2.26)
  - `changes-before` (New in version 2.66)
- For admin user, whitelist includes all filter keys mentioned in Request Section.

Normal response codes: 200

Error response codes: badRequest(400), unauthorized(401), forbidden(403)

Request¶

Name	In	Type	Description
access_ip_v4 (Optional)	query	string	Filter server list result by IPv4 address that should be used to access the server.
access_ip_v6 (Optional)	query	string	Filter server list result by IPv6 address that should be used to access the server.
all_tenants (Optional)	query	boolean	Specify the `all_tenants` query parameter to list all instances for all projects. By default this is only allowed by administrators. If the value of this parameter is not specified, it is treated as `True`. If
auto_disk_config (Optional)	query	string	Filter the server list result by the `disk_config` setting of the server, Valid values are:`AUTO``MANUAL`This parameter is only valid when specified by administrators. If non-admin users specify this p
availability_zone (Optional)	query	string	Filter the server list result by server availability zone.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
changes-since (Optional)	query	string	Filters the response by a date and time stamp when the server last changed status. To help keep track of changes this may also return recently deleted servers.The date and time stamp format is ISO specified, the value of the `changes-since` must be earlier than or equal to the value of the `changes-before` otherwise API will return 400.
config_drive (Optional)	query	string	Filter the server list result by the config drive setting of the server.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
created_at (Optional)	query	string	Filter the server list result by a date and time stamp when server was created.The date and time stamp format is ISO 8601:`CCYY-MM-DDThh:mm:ss±hh:mm`The `±hh:mm` value, if included, returns th
deleted (Optional)	query	boolean	Show deleted items only. In some circumstances deleted items will still be accessible via the backend database, however there is no contract on how long, so this parameter should be used with cau
description (Optional)	query	string	Filter the server list result by description.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored. Note`display_description` can als
flavor (Optional)	query	string	Filters the response by a flavor, as a UUID. A flavor is a combination of memory, disk size, and CPUs.
host (Optional)	query	string	Filter the server list result by the host name of compute node.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
hostname (Optional)	query	string	Filter the server list result by the host name of server.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
image (Optional)	query	string	Filters the response by an image, as a UUID. Note'image_ref' can also be requested which is alias of 'image' but that is not recommended to use as that will be removed in future.
ip (Optional)	query	string	An IPv4 address to filter results by.
ip6 (Optional)	query	string	An IPv6 address to filter results by.Up to microversion 2.4, this parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored. Starting from microv
kernel_id (Optional)	query	string	Filter the server list result by the UUID of the kernel image when using an AMI.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
key_name (Optional)	query	string	Filter the server list result by keypair name.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
launch_index (Optional)	query	integer	Filter the server list result by the sequence in which the servers were launched.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
launched_at (Optional)	query	string	Filter the server list result by a date and time stamp when the instance was launched. The date and time stamp format is ISO 8601:`CCYY-MM-DDThh:mm:ss±hh:mm`The `±hh:mm` value, if included, re
limit (Optional)	query	integer	Requests a page size of items. Returns a number of items up to a limit value. Use the `limit` parameter to make an initial limited request and use the ID of the last-seen item from the response as t
locked_by (Optional)	query	string	Filter the server list result by who locked the server, possible value could be `admin` or `owner`.This parameter is only valid when specified by administrators. If non-admin users specify this paramete
marker (Optional)	query	string	The ID of the last-seen item. Use the `limit` parameter to make an initial limited request and use the ID of the last-seen item from the response as the `marker` parameter value in a subsequent lim
name (Optional)	query	string	Filters the response by a server name, as a string. You can use regular expressions in the query. For example, the `?name=bob` regular expression returns both bob and bobb. If you must match on c removed in future.
node (Optional)	query	string	Filter the server list result by the node.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
power_state (Optional)	query	integer	Filter the server list result by server power state.Possible values are integer values that is mapped as:`0: NOSTATE 1: RUNNING 3: PAUSED 4: SHUTDOWN 6: CRASHED 7: SUSPENDED`This par
progress (Optional)	query	integer	Filter the server list result by the progress of the server. The value could be from 0 to 100 as integer.This parameter is only valid when specified by administrators. If non-admin users specify this par
project_id (Optional)	query	string	Filter the list of servers by the given project ID.This filter only works when the `all_tenants` filter is also specified. Note'tenant_id' can also be requested which is alias of 'project_id' but that is not re
ramdisk_id (Optional)	query	string	Filter the server list result by the UUID of the ramdisk image when using an AMI.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored
reservation_id (Optional)	query	string	A reservation id as returned by a servers multiple create call.
root_device_name (Optional)	query	string	Filter the server list result by the root device name of the server.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
soft_deleted (Optional)	query	boolean	Filter the server list by `SOFT_DELETED` status. This parameter is only valid when the `deleted=True` filter parameter is specified.
sort_dir (Optional)	query	string	Sort direction. A valid value is `asc` (ascending) or `desc` (descending). Default is `desc`. You can specify multiple pairs of sort key and sort direction query parameters. If you omit the sort direction in
sort_key (Optional)	query	string	Sorts by a server attribute. Default attribute is `created_at`. You can specify multiple pairs of sort key and sort direction query parameters. If you omit the sort direction in a pair, the API uses the na to:`access_ip_v4``access_ip_v6``auto_disk_config``availability_zone``config_drive``created_at``display_description``display_name``host``hostname``image_r and `node` are only allowed for admin. If non-admin users specify them, a 403 error is returned.
status (Optional)	query	string	Filters the response by a server status, as a string. For example, `ACTIVE`.Up to microversion 2.37, an empty list is returned if an invalid status is specified. Starting from microversion 2.38, a 400 erro
task_state (Optional)	query	string	Filter the server list result by task state.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
terminated_at (Optional)	query	string	Filter the server list result by a date and time stamp when instance was terminated. The date and time stamp format is ISO 8601:`CCYY-MM-DDThh:mm:ss±hh:mm`The `±hh:mm` value, if included, retu
user_id (Optional)	query	string	Filter the list of servers by the given user ID.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
uuid (Optional)	query	string	Filter the server list result by the UUID of the server.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
vm_state (Optional)	query	string	Filter the server list result by vm state.The value could be:`ACTIVE``BUILDING``DELETED``ERROR``PAUSED``RESCUED``RESIZED``SHELVED``SHELVED_OFFLOADED``SOFT_DELETED``STOPP
not-tags (Optional)	query	string	A list of tags to filter the server list by. Servers that don't match all tags in this list will be returned. Boolean expression in this case is 'NOT (t1 AND t2)'. Tags in query must be separated by comma.**Ne**
not-tags-any (Optional)	query	string	A list of tags to filter the server list by. Servers that don't match any tags in this list will be returned. Boolean expression in this case is 'NOT (t1 OR t2)'. Tags in query must be separated by comma.**Ne**
tags (Optional)	query	string	A list of tags to filter the server list by. Servers that match all tags in this list will be returned. Boolean expression in this case is 't1 AND t2'. Tags in query must be separated by comma.**New in versio**
tags-any (Optional)	query	string	A list of tags to filter the server list by. Servers that match any tag in this list will be returned. Boolean expression in this case is 't1 OR t2'. Tags in query must be separated by comma.**New in version**
changes-before (Optional)	query	string	Filters the response by a date and time stamp when the server last changed. Those servers that changed before or equal to the specified date and time stamp are returned. To help keep track of cha UTC time zone is assumed. When both `changes-since` and `changes-before` are specified, the value of the `changes-before` must be later than or equal to the value of the `changes-since` o

Response¶

Name	In	Type	Description
servers	body	array	A list of `server` objects.
id	body	string	The UUID of the server.
links	body	array	Links to the resources in question. See API Guide / Links and Referencesfor more info.
name	body	string	The server name.
servers_links (Optional)	body	array	Links to the next server. It is available when the number of servers exceeds `limit` parameter or `[api]/max_limit` in the configuration file. See API Guide / Links and References for more info.

Example List Servers


```
{
    "servers": [
        {
            "id": "22c91117-08de-4894-9aa9-6ef382400985",
            "links": [
                {
                    "href":
"http://openstack.example.com/v2/6f70656e737461636b20342065766572/servers/22c91117-08de-4894-9aa9-
6ef382400985",
                    "rel": "self"
                },
                {
                    "href":
"http://openstack.example.com/6f70656e737461636b20342065766572/servers/22c91117-08de-4894-9aa9-
6ef382400985",
                    "rel": "bookmark"
                }
            ],
            "name": "new-server-test"
        }
    ],
    "servers_links": [
        {
            "href": "http://openstack.example.com/v2.1/6f70656e737461636b20342065766572/servers?
limit=1&marker=22c91117-08de-4894-9aa9-6ef382400985",
            "rel": "next"
        }
    ]
}
```


Create Server POST /servers

Creates a server.

The progress of this operation depends on the location of the requested image, network I/O, host load, selected flavor, and other factors.

To check the progress of the request, make a `GET /servers/{id}` request. This call returns a progress attribute, which is a percentage value from 0 to 100.

The `Location` header returns the full URL to the newly created server and is available as a `self` and `bookmark` link in the server representation.

When you create a server, the response shows only the server ID, its links, and the admin password. You can get additional attributes through subsequent `GET` requests on the server.

Include the `block_device_mapping_v2` parameter in the create request body to boot a server from a volume.

Include the `key_name` parameter in the create request body to add a keypair to the server when you create it. To create a keypair, make a create keypair request.

> Note
>
> Starting with microversion 2.37 the `networks` field is required.

Preconditions

- The user must have sufficient server quota to create the number of servers requested.
- The connection to the Image service is valid.

Asynchronous postconditions

- With correct permissions, you can see the server status as `ACTIVE` through API calls.
- With correct access, you can see the created server in the compute node that OpenStack Compute manages.

Troubleshooting

- If the server status remains `BUILDING` or shows another error status, the request failed. Ensure you meet the preconditions then investigate the compute node.
- The server is not created in the compute node that OpenStack Compute manages.
- The compute node needs enough free resource to match the resource of the server creation request.
- Ensure that the scheduler selection filter can fulfill the request with the available compute nodes that match the selection criteria of the filter.

Normal response codes: 202

Error response codes: badRequest(400), unauthorized(401), forbidden(403), itemNotFound(404), conflict(409)

Request¶

Name	In	Type	Description
server	body	object	A `server` object.
flavorRef	body	string	The flavor reference, as an ID (including a UUID) or full URL, for the flavor for your server instance.
name	body	string	The server name.
networks	body	array	A list of `network` object. Required parameter when there are multiple networks defined for the tenant. When you do not specify the networks parameter, the server attaches to the only network created for the current tenant. Optionally, you can create one or more NICs on the server. To provision the server instance with a NIC for a network, specify the UUID of the network in the `uuid` attribute in a `networks` object. To provision the server instance with a NIC for an already existing port, specify the port-id in the `port` attribute in a `networks` object.If multiple networks are defined, the order in which they appear in the guest operating system will not necessarily reflect the order in which they are given in the server boot request. Guests should therefore not depend on device order to deduce any information about their network devices. Instead, device role tags should be used: introduced in 2.32, broken in 2.37, and re-introduced and fixed in 2.42, the `tag` is an optional, string attribute that can be used to assign a tag to a virtual network interface. This tag is then exposed to the guest in the metadata API and the config drive and is associated to hardware metadata for that network interface, such as bus (ex: PCI), bus address (ex: 0000:00:02.0), and MAC address.A bug has caused the `tag` attribute to no longer be accepted starting with version 2.37. Therefore, network interfaces could only be tagged in versions 2.32 to 2.36 inclusively. Version 2.42 has restored the `tag` attribute.Starting with microversion 2.37, this field is required and the special string values *auto* and *none* can be specified for networks. *auto* tells the Compute service to use a network that is available to the project, if one exists. If one does not exist, the Compute service will attempt to automatically allocate a network for the project (if possible). *none* tells the Compute service to not allocate a network for the instance. The *auto* and *none* values cannot be used with any other network values, including other network uuids, ports, fixed IPs or device tags. These are requested as strings for the networks value, not in a list. See the associated example.
networks.uuid (Optional)	body	string	To provision the server instance with a NIC for a network, specify the UUID of the network in the `uuid` attribute in a `networks` object. Required if you omit the `port` attribute.Starting with microversion 2.37, this value is strictly enforced to be in UUID format.
networks.port (Optional)	body	string	To provision the server instance with a NIC for an already existing port, specify the port-id in the `port` attribute in a `networks` object. The port status must be `DOWN`. Required if you omit the `uuid` attribute. Requested security groups are not applied to pre-existing ports.
networks.fixed_ip (Optional)	body	string	A fixed IPv4 address for the NIC. Valid with a `neutron` or `nova-networks` network.
networks.tag (Optional)	body	string	A device role tag that can be applied to a network interface. The guest OS of a server that has devices tagged in this manner can access hardware metadata about the tagged devices from the metadata API and on the config drive, if enabled. NoteDue to a bug, network interface tags are accepted between 2.32 and 2.36 inclusively, and subsequently starting with version 2.42.**New in version 2.32**
accessIPv4 (Optional)	body	string	IPv4 address that should be used to access this server.
accessIPv6 (Optional)	body	string	IPv6 address that should be used to access this server.
adminPass (Optional)	body	string	The administrative password of the server. If you omit this parameter, the operation generates a new password.
availability_zone (Optional)	body	string	The availability zone from which to launch the server. When you provision resources, you specify from which availability zone you want your instance to be built. Typically, an admin user will use availability zones to arrange OpenStack compute hosts into logical groups. An availability zone provides a form of physical isolation and redundancy from other availability zones. For instance, if some racks in your data center are on a separate power source, you can put servers in those racks in their own availability zone. Availability zones can also help separate different classes of hardware. By segregating resources into availability zones, you can ensure that your application resources are spread across disparate machines to achieve high availability in the event of hardware or other failure. You can list the available availability zones by calling the os-availability-zone API, but you should avoid using the default availability zone when booting the instance. In general, the default availability zone is named `nova`. This AZ is only shown when listing the availability zones as an admin.
block_device_mapping_v2 (Optional)	body	array	Enables fine grained control of the block device mapping for an instance. This is typically used for booting servers from volumes. An example format would look as follows: `"block_device_mapping_v2": [{ "boot_index": "0", "uuid": "ac408821-c95a-448f-9292-73986c790911", "source_type": "image", "volume_size": "25", "destination_type": "volume", "delete_on_termination": true, "tag": "disk1", "disk_bus": "scsi"}]` In microversion 2.32, `tag` is an optional string attribute that can be used to assign a tag to the block device. This tag is then exposed to the guest in the metadata API and the config drive and is associated to hardware metadata for that block device, such as bus (ex: SCSI), bus address (ex: 1:0:2:0), and serial.A bug has caused the `tag` attribute to no longer be accepted starting with version 2.33. It has been restored in version 2.42.
block_device_mapping_v2.boot_index	body	integer	Defines the order in which a hypervisor tries devices when it attempts to boot the guest from storage. Give each device a unique boot index starting from `0`. To disable a device from booting, set the boot index to a negative value or use the default boot index value, which is `None`. The simplest usage is, set the boot index of the boot device to `0` and use the default boot index value, `None`, for any other devices. Some hypervisors might not support booting from multiple devices; these hypervisors consider only the device with a boot index of `0`. Some hypervisors support booting from multiple devices but only if the devices are of different types. For example, a disk and CD-ROM.
block_device_mapping_v2.delete_on_termination (Optional)	body	boolean	To delete the boot volume when the server is destroyed, specify `true`. Otherwise, specify `false`. Default: `false`
block_device_mapping_v2.destination_type (Optional)	body	string	Defines where the block device mapping will reside. Valid values are: `local`: The ephemeral disk resides local to the compute host on which the server runs `volume`: The persistent volume is stored in the block storage service
block_device_mapping_v2.device_name (Optional)	body	string	A path to the device for the volume that you want to use to boot the server. Note that as of the 12.0.0 Liberty release, the Nova libvirt driver no longer honors a user-supplied device name. This is the same behavior as if the device name parameter is not supplied on the request.
block_device_mapping_v2.device_type (Optional)	body	string	The device type. For example, `disk`, `cdrom`.
block_device_mapping_v2.disk_bus (Optional)	body	string	Disk bus type, some hypervisors (currently only libvirt) support specify this parameter. Some example disk_bus values can be: ide, usb, virtio, scsi. This is not an exhaustive list as it depends on the virtualization driver, and may change as more support is added.
block_device_mapping_v2.guest_format (Optional)	body	string	Specifies the guest server disk file system format, such as `ext2`, `ext3`, `ext4`, `xfs` or `swap`.Swap block device mappings have the following restrictions:The `source_type` must be `blank`.The `destination_type` must be `local`.There can only be one swap disk per serverThe size of the swap disk must be less than or equal to the `swap` size of the flavor
block_device_mapping_v2.no_device (Optional)	body	boolean	It is no device if `True`.
			The source type of the block device. Valid values are: `blank`: Depending on the `destination_type` and

block_device_mapping_v2.source_type (Optional)	body	string	`guest_format`, this will either be a blank persistent volume or an ephemeral (or swap) disk local to the compute host on which the server resides `image`: This is only valid with `destination_type=volume`; creates an image-backed volume in the block storage service and attaches it to the server `snapshot`: This is only valid with `destination_type=volume`; creates a volume backed by the given volume snapshot referenced via the `block_device_mapping_v2.uuid` parameter and attaches it to the server `volume`: This is only valid with `destination_type=volume`; uses the existing persistent volume referenced via the `block_device_mapping_v2.uuid` parameter and attaches it to the serverThis parameter is required unless `block_device_mapping_v2.no_device` is specified.See Block Device Mapping in Nova for more details on valid source and destination types.
block_device_mapping_v2.uuid (Optional)	body	string	This is the uuid of source resource. The uuid points to different resources based on the `source_type`. For example, if `source_type` is `image`, the block device is created based on the specified image which is retrieved from the image service. Similarly, if `source_type` is `snapshot` then the uuid refers to a volume snapshot in the block storage service. If `source_type` is `volume` then the uuid refers to a volume in the block storage service.
block_device_mapping_v2.volume_size (Optional)	body	integer	The size of the volume (in GiB). This is integer value from range 1 to 2147483647 which can be requested as integer and string.
block_device_mapping_v2.tag (Optional)	body	string	A device role tag that can be applied to a block device. The guest OS of a server that has devices tagged in this manner can access hardware metadata about the tagged devices from the metadata API and on the config drive, if enabled. NoteDue to a bug, block device tags are accepted in version 2.32 and subsequently starting with version 2.42.**New in version 2.32**
block_device_mapping_v2.volume_type (Optional)	body	string	The device `volume_type`. This can be used to specify the type of volume which the compute service will create and attach to the server. If not specified, the block storage service will provide a default volume type. See the block storage volume types API for more details. There are some restrictions on `volume_type`:It can be a volume type ID or name.It is only supported with `source_type` of `blank`, `image` or `snapshot`.It is only supported with `destination_type` of `volume`.**New in version 2.67**
config_drive (Optional)	body	boolean	Indicates whether a configuration drive enables metadata injection. The config_drive setting provides information about a drive that the instance can mount at boot time. The instance reads files from the drive to get information that is normally available through the metadata service. This metadata is different from the user data. Not all cloud providers enable the `config_drive`. Read more in the OpenStack End User Guide.
imageRef (Optional)	body	string	The UUID of the image to use for your server instance. This is not required in case of boot from volume. In all other cases it is required and must be a valid UUID otherwise API will return 400.
key_name (Optional)	body	string	Key pair name. NoteThe `null` value was allowed in the Nova legacy v2 API, but due to strict input validation, it is not allowed in the Nova v2.1 API.
metadata (Optional)	body	object	Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each.
OS-DCF:diskConfig (Optional)	body	string	Controls how the API partitions the disk when you create, rebuild, or resize servers. A server inherits the `OS-DCF:diskConfig` value from the image from which it was created, and an image inherits the `OS-DCF:diskConfig` value from the server from which it was created. To override the inherited setting, you can include this attribute in the request body of a server create, rebuild, or resize request. If the `OS-DCF:diskConfig` value for an image is `MANUAL`, you cannot create a server from that image and set its `OS-DCF:diskConfig` value to `AUTO`. A valid value is: `AUTO`. The API builds the server with a single partition the size of the target flavor disk. The API automatically adjusts the file system to fit the entire partition. `MANUAL`. The API builds the server by using whatever partition scheme and file system is in the source image. If the target flavor disk is larger, the API does not partition the remaining disk space.
personality (Optional)	body	array	The file path and contents, text only, to inject into the server at launch. The maximum size of the file path data is 255 bytes. The maximum limit is the number of allowed bytes in the decoded, rather than encoded, data.**Available until version 2.56**
security_groups (Optional)	body	array	One or more security groups. Specify the name of the security group in the `name` attribute. If you omit this attribute, the API creates the server in the `default` security group. Requested security groups are not applied to pre-existing ports.
user_data (Optional)	body	string	Configuration information or scripts to use upon launch. Must be Base64 encoded. Restricted to 65535 bytes. NoteThe `null` value allowed in Nova legacy v2 API, but due to the strict input validation, it isn't allowed in Nova v2.1 API.
description (Optional)	body	string	A free form description of the server. Limited to 255 characters in length. Before microversion 2.19 this was set to the server name.**New in version 2.19**
tags (Optional)	body	array	A list of tags. Tags have the following restrictions:Tag is a Unicode bytestring no longer than 60 characters.Tag is a non-empty string.'/' is not allowed to be in a tag nameComma is not allowed to be in a tag name in order to simplify requests that specify lists of tagsAll other characters are allowed to be in a tag nameEach server can have up to 50 tags.**New in version 2.52**
trusted_image_certificates (Optional)	body	array	A list of trusted certificate IDs, which are used during image signature verification to verify the signing certificate. The list is restricted to a maximum of 50 IDs. This parameter is optional in server create requests if allowed by policy, and is not supported for volume-backed instances.**New in version 2.63**
os:scheduler_hints (Optional)	body	object	The dictionary of data to send to the scheduler. Alternatively, you can specify `OS-SCH-HNT:scheduler_hints` as the key in the request body. NoteThis is a top-level key in the request body, not part of the server portion of the request body.There are a few caveats with scheduler hints:The request validation schema is per hint. For example, some require a single string value, and some accept a list of values.Hints are only used based on the cloud scheduler configuration, which varies per deployment.Hints are pluggable per deployment, meaning that a cloud can have custom hints which may not be available in another cloud.For these reasons, it is important to consult each cloud's user documentation to know what is available for scheduler hints.
os:scheduler_hints.build_near_host_ip (Optional)	body	string	Schedule the server on a host in the network specified with this parameter and a cidr (`os:scheduler_hints.cidr`). It is available when `SimpleCIDRAffinityFilter` is available on cloud side.
os:scheduler_hints.cidr (Optional)	body	string	Schedule the server on a host in the network specified with an IP address (`os:scheduler_hints:build_near_host_ip`) and this parameter. If `os:scheduler_hints:build_near_host_ip` is specified and this paramete is omitted, `/24` is used. It is available when `SimpleCIDRAffinityFilter` is available on cloud side.
os:scheduler_hints.different_cell (Optional)	body	array	A list of cell routes or a cell route (string). Schedule the server in a cell that is not specified. It is available when `DifferentCellFilter` is available on cloud side that is cell v1 environment.
os:scheduler_hints.different_host (Optional)	body	array	A list of server UUIDs or a server UUID. Schedule the server on a different host from a set of servers. It is available when `DifferentHostFilter` is available on cloud side.
os:scheduler_hints.group (Optional)	body	string	The server group UUID. Schedule the server according to a policy of the server group (`anti-affinity`, `affinity`, `soft-anti-affinity` or `soft-affinity`). It is available when `ServerGroupAffinityFilter`, `ServerGroupAntiAffinityFilter`, `ServerGroupSoftAntiAffinityWeigher`, `ServerGroupSoftAffinityWeigher` are available on cloud side.
os:scheduler_hints.query (Optional)	body	string	Schedule the server by using a custom filter in JSON format. For example: `"query": "[>=,$free_ram_mb,1024]"` It is available when `JsonFilter` is available on cloud side.
os:scheduler_hints.same_host (Optional)	body	array	A list of server UUIDs or a server UUID. Schedule the server on the same host as another server in a set of servers. It is available when `SameHostFilter` is available on cloud side.
os:scheduler_hints.target_cell (Optional)	body	string	A target cell name. Schedule the server in a host in the cell specified. It is available when `TargetCellFilter` is available on cloud side that is cell v1 environment.

Example Create Server

```
{
    "server" : {
```

```
        "accessIPv4": "1.2.3.4",
        "accessIPv6": "80fe::",
        "name" : "new-server-test",
        "imageRef" : "70a599e0-31e7-49b7-b260-868f441e862b",
        "flavorRef" : "1",
        "availability_zone": "nova",
        "OS-DCF:diskConfig": "AUTO",
        "metadata" : {
            "My Server Name" : "Apache1"
        },
        "personality": [
            {
                "path": "/etc/banner.txt",
                "contents": "ICAgICAgDQoiQSBjbG91ZCBkb2VzIG5vdCBrbm93IHdoeSBp
dCBtb3ZlcyBpbiBqdXN0IHN1Y2ggYSBkaXJlY3Rpb24gYW5k IGF0IHN1Y2ggYSBzcGVlZC4uLklOIGZlZWxzIGFuIGltcHVs
c2lvbi4uLnRoaXMgaXMgdGhlIHBsYWNlIHRvIGdvIG5vdy4g QnVOIHRoZSBza3kga25vd3MgdGhlIHJlYXNvbnMgYW5kIHRo
ZSBwYXR0ZXJucyBiZWhpbmQgYWxsIGNsb3VkcywgYW5kIHlv dSB3aWxsIGtub3csIHRvbywgd2hlbiB5b3UgbGlmdCB5b3Vy
c2VsZiBoaWdoIGVub3VnaCBOObyBzZWUgYmV5b25kIGhvcml6 b25zLiINCgOKLVJpY2hhcmQgQmFjaA=="
            }
        ],
        "security_groups": [
            {
                "name": "default"
            }
        ],
        "user_data" : "IyEvYmluL2Jhc2gKL2Jpbi9zdQplY2hvICJJIGFtIGluIHlvdSEiCg=="
    },
    "OS-SCH-HNT:scheduler_hints": {
        "same_host": "48e6a9f6-30af-47e0-bc04-acaed113bb4e"
    }
}
```

Example Create Server With Networks(array) and Block Device Mapping V2 (v2.32)

```
{
    "server" : {
        "name" : "device-tagging-server",
        "flavorRef" : "http://openstack.example.com/flavors/1",
        "networks" : [{
            "uuid" : "ff608d40-75e9-48cb-b745-77bb55b5eaf2",
            "tag": "nic1"
        }],
        "block_device_mapping_v2": [{
            "uuid": "70a599e0-31e7-49b7-b260-868f441e862b",
            "source_type": "image",
            "destination_type": "volume",
            "boot_index": 0,
            "volume_size": "1",
            "tag": "disk1"
        }]
    }
}
```

Example Create Server With Automatic Networking (v2.37)

```
{
    "server": {
        "name": "auto-allocate-network",
        "imageRef": "70a599e0-31e7-49b7-b260-868f441e862b",
        "flavorRef": "http://openstack.example.com/flavors/1",
        "networks": "auto"
    }
}
```

Example Create Server With Trusted Image Certificates (v2.63)

```
{
    "server" : {
        "accessIPv4": "1.2.3.4",
        "accessIPv6": "80fe::",
        "name" : "new-server-test",
        "imageRef" : "70a599e0-31e7-49b7-b260-868f441e862b",
        "flavorRef" : "6",
        "availability_zone": "nova",
        "OS-DCF:diskConfig": "AUTO",
        "metadata" : {
            "My Server Name" : "Apache1"
        },
```

```
        "security_groups": [
            {
                "name": "default"
            }
        ],
        "user_data" : "IyEvYmluL2Jhc2gKL2Jpbi9zdQplY2hvICJJIGFtIGluIHlvdSEiCg==",
        "networks": "auto",
        "trusted_image_certificates": [
            "0b5d2c72-12cc-4ba6-a8d7-3ff5cc1d8cb8",
            "674736e3-f25c-405c-8362-bbf991e0ce0a"
        ]
    },
    "OS-SCH-HNT:scheduler_hints": {
        "same_host": "[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}"
    }
}
```

Response¶

Name	In	Type	Description
Location	header	string	The location URL of the server, HTTP header “Location: ” will be returned.
server	body	object	A `server` object.
id	body	string	The UUID of the server.
links	body	array	Links to the resources in question. See API Guide / Links and Referencesfor more info.
OS-DCF:diskConfig	body	string	Disk configuration. The value is either: `AUTO`. The API builds the server with a single partition the size of the target flavor disk. The API automatically adjusts the file system to fit the entire partition. `MANUAL`. The API builds the server by using the partition scheme and file system that is in the source image. If the target flavor disk is larger, The API does not partition the remaining disk space.
security_groups	body	array	One or more security groups objects.
security_groups.name	body	string	The security group name.
adminPass (Optional)	body	string	The administrative password for the server. If you set `enable_instance_password` configuration option to `False`, the API wouldn’t return the `adminPass` field in response.

Example Create Server

```
{
    "server": {
        "OS-DCF:diskConfig": "AUTO",
        "adminPass": "6NpUwoz2QDRN",
        "id": "f5dc173b-6804-445a-a6d8-c705dad5b5eb",
        "links": [
            {
                "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/servers/f5dc173b-
6804-445a-a6d8-c705dad5b5eb",
                "rel": "self"
            },
            {
                "href": "http://openstack.example.com/6f70656e737461636b20342065766572/servers/f5dc173b-
6804-445a-a6d8-c705dad5b5eb",
                "rel": "bookmark"
            }
        ],
        "security_groups": [
            {
                "name": "default"
            }
        ]
    }
}
```

List Servers Detailed GET /servers/detail

For each server, shows server details including configuration drive, extended status, and server usage information.

The extended status information appears in the OS-EXT-STS:vm_state, OS-EXT-STS:power_state, and OS-EXT-STS:task_state attributes.

The server usage information appears in the OS-SRV-USG:launched_at and OS-SRV-USG:terminated_at attributes.

HostId is unique per account and is not globally unique.

Normal response codes: 200

Error response codes: badRequest(400), unauthorized(401), forbidden(403)

Request¶

Name	In	Type	Description
access_ip_v4 (Optional)	query	string	Filter server list result by IPv4 address that should be used to access the server.
access_ip_v6 (Optional)	query	string	Filter server list result by IPv6 address that should be used to access the server.
all_tenants (Optional)	query	boolean	Specify the `all_tenants` query parameter to list all instances for all projects. By default this is only allowed by administrators. If the value of this parameter is not specified, it is treated as `True`. If
auto_disk_config (Optional)	query	string	Filter the server list result by the `disk_config` setting of the server, Valid values are:`AUTO``MANUAL`This parameter is only valid when specified by administrators. If non-admin users specify this p
availability_zone (Optional)	query	string	Filter the server list result by server availability zone.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
changes-since (Optional)	query	string	Filters the response by a date and time stamp when the server last changed status. To help keep track of changes this may also return recently deleted servers.The date and time stamp format is ISO specified, the value of the `changes-since` must be earlier than or equal to the value of the `changes-before` otherwise API will return 400.
config_drive (Optional)	query	string	Filter the server list result by the config drive setting of the server.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
created_at (Optional)	query	string	Filter the server list result by a date and time stamp when server was created.The date and time stamp format is ISO 8601:`CCYY-MM-DDThh:mm:ss±hh:mm`The `±hh:mm` value, if included, returns th
deleted (Optional)	query	boolean	Show deleted items only. In some circumstances deleted items will still be accessible via the backend database, however there is no contract on how long, so this parameter should be used with cau
description (Optional)	query	string	Filter the server list result by description.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored. Note`display_description` can als
flavor (Optional)	query	string	Filters the response by a flavor, as a UUID. A flavor is a combination of memory, disk size, and CPUs.
host (Optional)	query	string	Filter the server list result by the host name of compute node.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
hostname (Optional)	query	string	Filter the server list result by the host name of server.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
image (Optional)	query	string	Filters the response by an image, as a UUID. Note'image_ref' can also be requested which is alias of 'image' but that is not recommended to use as that will be removed in future.
ip (Optional)	query	string	An IPv4 address to filter results by.
ip6 (Optional)	query	string	An IPv6 address to filter results by.Up to microversion 2.4, this parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored. Starting from microv
kernel_id (Optional)	query	string	Filter the server list result by the UUID of the kernel image when using an AMI.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
key_name (Optional)	query	string	Filter the server list result by keypair name.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
launch_index (Optional)	query	integer	Filter the server list result by the sequence in which the servers were launched.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
launched_at (Optional)	query	string	Filter the server list result by a date and time stamp when the instance was launched. The date and time stamp format is ISO 8601:`CCYY-MM-DDThh:mm:ss±hh:mm`The `±hh:mm` value, if included, re
limit (Optional)	query	integer	Requests a page size of items. Returns a number of items up to a limit value. Use the `limit` parameter to make an initial limited request and use the ID of the last-seen item from the response as t
locked_by (Optional)	query	string	Filter the server list result by who locked the server, possible value could be `admin` or `owner`.This parameter is only valid when specified by administrators. If non-admin users specify this paramete
marker (Optional)	query	string	The ID of the last-seen item. Use the `limit` parameter to make an initial limited request and use the ID of the last-seen item from the response as the `marker` parameter value in a subsequent lim
name (Optional)	query	string	Filters the response by a server name, as a string. You can use regular expressions in the query. For example, the `?name=bob` regular expression returns both bob and bobb. If you must match on c removed in future.
node (Optional)	query	string	Filter the server list result by the node.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
power_state (Optional)	query	integer	Filter the server list result by server power state.Possible values are integer values that is mapped as:`0: NOSTATE 1: RUNNING 3: PAUSED 4: SHUTDOWN 6: CRASHED 7: SUSPENDED`This par
progress (Optional)	query	integer	Filter the server list result by the progress of the server. The value could be from 0 to 100 as integer.This parameter is only valid when specified by administrators. If non-admin users specify this par
project_id (Optional)	query	string	Filter the list of servers by the given project ID.This filter only works when the `all_tenants` filter is also specified. Note'tenant_id' can also be requested which is alias of 'project_id' but that is not re
ramdisk_id (Optional)	query	string	Filter the server list result by the UUID of the ramdisk image when using an AMI.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored
reservation_id (Optional)	query	string	A reservation id as returned by a servers multiple create call.
root_device_name (Optional)	query	string	Filter the server list result by the root device name of the server.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
soft_deleted (Optional)	query	boolean	Filter the server list by `SOFT_DELETED` status. This parameter is only valid when the `deleted=True` filter parameter is specified.
sort_dir (Optional)	query	string	Sort direction. A valid value is `asc` (ascending) or `desc` (descending). Default is `desc`. You can specify multiple pairs of sort key and sort direction query parameters. If you omit the sort direction in
sort_key (Optional)	query	string	Sorts by a server attribute. Default attribute is `created_at`. You can specify multiple pairs of sort key and sort direction query parameters. If you omit the sort direction in a pair, the API uses the na to:`access_ip_v4``access_ip_v6``auto_disk_config``availability_zone``config_drive``created_at``display_description``display_name``host``hostname``image_r and `node` are only allowed for admin. If non-admin users specify them, a 403 error is returned.
status (Optional)	query	string	Filters the response by a server status, as a string. For example, `ACTIVE`.Up to microversion 2.37, an empty list is returned if an invalid status is specified. Starting from microversion 2.38, a 400 erro
task_state (Optional)	query	string	Filter the server list result by task state.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
terminated_at (Optional)	query	string	Filter the server list result by a date and time stamp when instance was terminated. The date and time stamp format is ISO 8601:`CCYY-MM-DDThh:mm:ss±hh:mm`The `±hh:mm` value, if included, retu
user_id (Optional)	query	string	Filter the list of servers by the given user ID.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
uuid (Optional)	query	string	Filter the server list result by the UUID of the server.This parameter is only valid when specified by administrators. If non-admin users specify this parameter, it is ignored.
vm_state (Optional)	query	string	Filter the server list result by vm state.The value could be:`ACTIVE``BUILDING``DELETED``ERROR``PAUSED``RESCUED``RESIZED``SHELVED``SHELVED_OFFLOADED``SOFT_DELETED``STOPP
not-tags (Optional)	query	string	A list of tags to filter the server list by. Servers that don't match all tags in this list will be returned. Boolean expression in this case is 'NOT (t1 AND t2)'. Tags in query must be separated by comma.**Ne**
not-tags-any (Optional)	query	string	A list of tags to filter the server list by. Servers that don't match any tags in this list will be returned. Boolean expression in this case is 'NOT (t1 OR t2)'. Tags in query must be separated by comma.**Ne**
tags (Optional)	query	string	A list of tags to filter the server list by. Servers that match all tags in this list will be returned. Boolean expression in this case is 't1 AND t2'. Tags in query must be separated by comma.**New in versio**
tags-any (Optional)	query	string	A list of tags to filter the server list by. Servers that match any tag in this list will be returned. Boolean expression in this case is 't1 OR t2'. Tags in query must be separated by comma.**New in version**
changes-before (Optional)	query	string	Filters the response by a date and time stamp when the server last changed. Those servers that changed before or equal to the specified date and time stamp are returned. To help keep track of cha UTC time zone is assumed. When both `changes-since` and `changes-before` are specified, the value of the `changes-before` must be later than or equal to the value of the `changes-since` o

Response¶

Name	In	Type	Description
server	body	object	A `server` object.
accessIPv4	body	string	IPv4 address that should be used to access this server. May be automatically set by the provider.
accessIPv6	body	string	IPv6 address that should be used to access this server. May be automatically set by the provider.
addresses	body	object	The addresses for the server. Servers with status `BUILD` hide their addresses information.
config_drive	body	string	Indicates whether or not a config drive was used for this server. The value is `True` or an empty string. An empty string stands for `False`.
created	body	string	The date and time when the resource was created. The date and time stamp format is ISO 8601 `CCYY-MM-DDThh:mm:ss±hh:mm` For example, `2015-08-27T09:49:58-05:00`. The `±hh:mm` value, if included, is the time zone as an offset from UTC. In the previous example, the offset value is `-05:00`.
flavor	body	object	Before microversion 2.47 this contains the ID and links for the flavor used to boot the server instance. This

flavor	body	object	can be an empty object in case flavor information is no longer present in the system.As of microversion 2.47 this contains a subset of the actual flavor information used to create the server instance, represented as a nested dictionary.
flavor.id	body	string	The ID of the flavor. While people often make this look like an int, this is really a string.**Available until version 2.46**
flavor.links	body	array	Links to the flavor resource. See API Guide / Links and References for more info.**Available until version 2.46**
flavor.vcpus	body	integer	The number of virtual CPUs that were allocated to the server.**New in version 2.47**
flavor.ram	body	integer	The amount of RAM a flavor has, in MiB.**New in version 2.47**
flavor.disk	body	integer	The size of the root disk that was created in GiB.**New in version 2.47**
flavor.ephemeral	body	integer	The size of the ephemeral disk that was created, in GiB.**New in version 2.47**
flavor.swap	body	integer	The size of a dedicated swap disk that was allocated, in MiB.**New in version 2.47**
flavor.original_name	body	string	The display name of a flavor.**New in version 2.47**
flavor.extra_specs (Optional)	body	object	A dictionary of the flavor's extra-specs key-and-value pairs. This will only be included if the user is allowed by policy to index flavor extra_specs.**New in version 2.47**
flavor.extra_specs.key	body	string	The extra spec key of a flavor.**New in version 2.47**
flavor.extra_specs.value	body	string	The extra spec value of a flavor.**New in version 2.47**
hostId	body	string	An ID string representing the host. This is a hashed value so will not actually look like a hostname, and is hashed with data from the project_id, so the same physical host as seen by two different project_ids, will be different. It is useful when within the same project you need to determine if two instances are on the same or different physical hosts for the purposes of availability or performance.
id	body	string	The UUID of the server.
image	body	object	The UUID and links for the image for your server instance. The `image` object might be an empty string when you boot the server from a volume.
key_name	body	string	The name of associated key pair, if any.
links	body	array	Links to the resources in question. See API Guide / Links and References for more info.
metadata	body	object	A dictionary of metadata key-and-value pairs, which is maintained for backward compatibility.
name	body	string	The server name.
OS-DCF:diskConfig	body	string	Disk configuration. The value is either: `AUTO`. The API builds the server with a single partition the size of the target flavor disk. The API automatically adjusts the file system to fit the entire partition. `MANUAL`. The API builds the server by using the partition scheme and file system that is in the source image. If the target flavor disk is larger, The API does not partition the remaining disk space.
OS-EXT-AZ:availability_zone	body	string	The availability zone name.
OS-EXT-SRV-ATTR:host	body	string	The name of the compute host on which this instance is running. Appears in the response for administrative users only.
OS-EXT-SRV-ATTR:hypervisor_hostname	body	string	The hypervisor host name provided by the Nova virt driver. For the Ironic driver, it is the Ironic node uuid. Appears in the response for administrative users only.
OS-EXT-SRV-ATTR:instance_name	body	string	The instance name. The Compute API generates the instance name from the instance name template. Appears in the response for administrative users only.
OS-EXT-STS:power_state	body	integer	The power state of the instance. This is an enum value that is mapped as: `0: NOSTATE 1: RUNNING 3: PAUSED 4: SHUTDOWN 6: CRASHED 7: SUSPENDED`
OS-EXT-STS:task_state	body	string	The task state of the instance.
OS-EXT-STS:vm_state	body	string	The VM state.
os-extended-volumes:volumes_attached	body	array	The attached volumes, if any.
os-extended-volumes:volumes_attached.id	body	string	The attached volume ID.
os-extended-volumes:volumes_attached.delete_on_termination	body	boolean	A flag indicating if the attached volume will be deleted when the server is deleted. By default this is False and can only be set when creating a volume while creating a server, which is commonly referred to as boot from volume.**New in version 2.3**
OS-SRV-USG:launched_at	body	string	The date and time when the server was launched.The date and time stamp format is ISO 8601: `CCYY-MM-DDThh:mm:ss±hh:mm` For example, `2015-08-27T09:49:58-05:00`.The `hh±:mm` value, if included, is the time zone as an offset from UTC. If the `deleted_at` date and time stamp is not set, its value is `null`.
OS-SRV-USG:terminated_at	body	string	The date and time when the server was deleted.The date and time stamp format is ISO 8601: `CCYY-MM-DDThh:mm:ss±hh:mm` For example, `2015-08-27T09:49:58-05:00`. The `±hh:mm` value, if included, is the time zone as an offset from UTC. If the `deleted_at` date and time stamp is not set, its value is `null`.
security_groups	body	array	One or more security groups objects.
security_group.name	body	string	The security group name.
status	body	string	The server status.
tenant_id	body	string	The UUID of the tenant in a multi-tenancy cloud.
updated	body	string	The date and time when the resource was updated. The date and time stamp format is ISO 8601 `CCYY-MM-DDThh:mm:ss±hh:mm` For example, `2015-08-27T09:49:58-05:00`. The `±hh:mm` value, if included, is the time zone as an offset from UTC. In the previous example, the offset value is `-05:00`.
user_id	body	string	The user ID of the user who owns the server.
fault (Optional)	body	object	A fault object. Only displayed when the server status is `ERROR` or `DELETED` and a fault occurred.
fault.code	body	integer	The error response code.
fault.created	body	string	The date and time when the exception was raised. The date and time stamp format is ISO 8601 `CCYY-MM-DDThh:mm:ss±hh:mm` For example, `2015-08-27T09:49:58-05:00`. The `±hh:mm` value, if included, is the time zone as an offset from UTC. In the previous example, the offset value is `-05:00`.
fault.message	body	string	The error message.
fault.details (Optional)	body	string	The stack trace. It is available if the response code is not 500 or you have the administrator privilege
progress (Optional)	body	integer	A percentage value of the operation progress. This parameter only appears when the server status is `ACTIVE`, `BUILD`, `REBUILD`, `RESIZE`, `VERIFY_RESIZE` or `MIGRATING`.
servers_links (Optional)	body	array	Links to the next server. It is available when the number of servers exceeds `limit` parameter or `[api]/max_limit` in the configuration file. See API Guide / Links and References for more info.
OS-EXT-SRV-ATTR:hostname (Optional)	body	string	The hostname set on the instance when it is booted. By default, it appears in the response for administrative users only.**New in version 2.3**
OS-EXT-SRV-ATTR:reservation_id (Optional)	body	string	The reservation id for the server. This is an id that can be useful in tracking groups of servers created with multiple create, that will all have the same reservation_id. By default, it appears in the response for administrative users only.**New in version 2.3**

OS-EXT-SRV-ATTR:launch_index (Optional)	body	integer	When servers are launched via multiple create, this is the sequence in which the servers were launched. By default, it appears in the response for administrative users only.**New in version 2.3**
OS-EXT-SRV-ATTR:kernel_id (Optional)	body	string	The UUID of the kernel image when using an AMI. Will be null if not. By default, it appears in the response for administrative users only.**New in version 2.3**
OS-EXT-SRV-ATTR:ramdisk_id (Optional)	body	string	The UUID of the ramdisk image when using an AMI. Will be null if not. By default, it appears in the response for administrative users only.**New in version 2.3**
OS-EXT-SRV-ATTR:root_device_name (Optional)	body	string	The root device name for the instance By default, it appears in the response for administrative users only.**New in version 2.3**
OS-EXT-SRV-ATTR:user_data (Optional)	body	string	The user_data the instance was created with. By default, it appears in the response for administrative users only.**New in version 2.3**
locked	body	boolean	True if the instance is locked otherwise False.**New in version 2.9**
host_status (Optional)	body	string	The host status. Values where next value in list can override the previous: `UP` if nova-compute up. `UNKNOWN` if nova-compute not reported by servicegroup driver. `DOWN` if nova-compute forced down. `MAINTENANCE` if nova-compute is disabled.Empty string indicates there is no host for server.This attribute appears in the response only if the policy permits. By default, only administrators can get this parameter.**New in version 2.16**
description	body	string	The description of the server. Before microversion 2.19 this was set to the server name.**New in version 2.19**
tags	body	array	A list of tags. The maximum count of tags in this list is 50.**New in version 2.26**
trusted_image_certificates	body	array	A list of trusted certificate IDs, that were used during image signature verification to verify the signing certificate. The list is restricted to a maximum of 50 IDs. The value is `null` if trusted certificate IDs are not set.**New in version 2.63**

Example List Servers Detailed (2.63)


```
{
    "servers": [
        {
            "OS-DCF:diskConfig": "AUTO",
            "OS-EXT-AZ:availability_zone": "nova",
            "OS-EXT-SRV-ATTR:host": "compute",
            "OS-EXT-SRV-ATTR:hostname": "new-server-test",
            "OS-EXT-SRV-ATTR:hypervisor_hostname": "fake-mini",
            "OS-EXT-SRV-ATTR:instance_name": "instance-00000001",
            "OS-EXT-SRV-ATTR:kernel_id": "",
            "OS-EXT-SRV-ATTR:launch_index": 0,
            "OS-EXT-SRV-ATTR:ramdisk_id": "",
            "OS-EXT-SRV-ATTR:reservation_id": "r-y0w4v32k",
            "OS-EXT-SRV-ATTR:root_device_name": "/dev/sda",
            "OS-EXT-SRV-ATTR:user_data": "IyEvYmluL2Jhc2gKL2Jpbi9zdQplY2hvICJJIGFtIGluIHlvdSEiCg==",
            "OS-EXT-STS:power_state": 1,
            "OS-EXT-STS:task_state": null,
            "OS-EXT-STS:vm_state": "active",
            "OS-SRV-USG:launched_at": "2017-10-10T15:49:09.516729",
            "OS-SRV-USG:terminated_at": null,
            "accessIPv4": "1.2.3.4",
            "accessIPv6": "80fe::",
            "addresses": {
                "private": [
                    {
                        "OS-EXT-IPS-MAC:mac_addr": "aa:bb:cc:dd:ee:ff",
                        "OS-EXT-IPS:type": "fixed",
                        "addr": "192.168.0.3",
                        "version": 4
                    }
                ]
            },
            "config_drive": "",
            "created": "2017-10-10T15:49:08Z",
            "description": null,
            "flavor": {
                "disk": 1,
                "ephemeral": 0,
                "extra_specs": {
                    "hw:cpu_policy": "dedicated",
                    "hw:mem_page_size": "2048"
                },
                "original_name": "m1.tiny.specs",
                "ram": 512,
                "swap": 0,
                "vcpus": 1
            },
            "hostId": "2091634baaccdc4c5a1d57069c833e402921df696b7f970791b12ec6",
            "host_status": "UP",
            "id": "569f39f9-7c76-42a1-9c2d-8394e2638a6d",
            "image": {
                "id": "70a599e0-31e7-49b7-b260-868f441e862b",
                "links": [
                    {
```

```
                        "href":
"http://openstack.example.com/6f70656e737461636b20342065766572/images/70a599e0-31e7-49b7-b260-
868f441e862b",
                        "rel": "bookmark"
                    }
                ]
            },
            "key_name": null,
            "links": [
                {
                    "href":
"http://openstack.example.com/v2.1/6f70656e737461636b20342065766572/servers/569f39f9-7c76-42a1-9c2d-
8394e2638a6d",
                    "rel": "self"
                },
                {
                    "href":
"http://openstack.example.com/6f70656e737461636b20342065766572/servers/569f39f9-7c76-42a1-9c2d-
8394e2638a6d",
                    "rel": "bookmark"
                }
            ],
            "locked": false,
            "metadata": {
                "My Server Name": "Apache1"
            },
            "name": "new-server-test",
            "os-extended-volumes:volumes_attached": [],
            "progress": 0,
            "security_groups": [
                {
                    "name": "default"
                }
            ],
            "status": "ACTIVE",
            "tags": [],
            "tenant_id": "6f70656e737461636b20342065766572",
            "trusted_image_certificates": [
                "0b5d2c72-12cc-4ba6-a8d7-3ff5cc1d8cb8",
                "674736e3-f25c-405c-8362-bbf991e0ce0a"
            ],
            "updated": "2017-10-10T15:49:09Z",
            "user_id": "fake"
        }
    ],
    "servers_links": [
        {
            "href": "http://openstack.example.com/v2.1/6f70656e737461636b20342065766572/servers/detail?
limit=1&marker=569f39f9-7c76-42a1-9c2d-8394e2638a6d",
            "rel": "next"
        }
    ]
}
```


Show Server Details GET /servers/{server_id}

Shows details for a server.

Includes server details including configuration drive, extended status, and server usage information.

The extended status information appears in the `OS-EXT-STS:vm_state`, `OS-EXT-STS:power_state`, and `OS-EXT-STS:task_state` attributes.

The server usage information appears in the `OS-SRV-USG:launched_at` and `OS-SRV-USG:terminated_at` attributes.

HostId is unique per account and is not globally unique.

Preconditions

The server must exist.

Normal response codes: 200

Error response codes: unauthorized(401), forbidden(403), itemNotFound(404)

Request¶

Name	In	Type	Description
server_id	path	string	The UUID of the server.

Response¶

Name	In	Type	Description
server	body	object	A `server` object.
accessIPv4	body	string	IPv4 address that should be used to access this server. May be automatically set by the provider.
accessIPv6	body	string	IPv6 address that should be used to access this server. May be automatically set by the provider.
addresses	body	object	The addresses for the server. Servers with status `BUILD` hide their addresses information.
config_drive	body	string	Indicates whether or not a config drive was used for this server. The value is `True` or an empty string. An empty string stands for `False`.
created	body	string	The date and time when the resource was created. The date and time stamp format is ISO 8601 `CCYY-MM-DDThh:mm:ss±hh:mm` For example, `2015-08-27T09:49:58-05:00`. The `±hh:mm` value, if included, is the time zone as an offset from UTC. In the previous example, the offset value is `-05:00`.
flavor	body	object	Before microversion 2.47 this contains the ID and links for the flavor used to boot the server instance. This can be an empty object in case flavor information is no longer present in the system.As of microversion 2.47 this contains a subset of the actual flavor information used to create the server instance, represented as a nested dictionary.
flavor.id	body	string	The ID of the flavor. While people often make this look like an int, this is really a string.**Available until version 2.46**
flavor.links	body	array	Links to the flavor resource. See API Guide / Links and References for more info.**Available until version 2.46**
flavor.vcpus	body	integer	The number of virtual CPUs that were allocated to the server.**New in version 2.47**
flavor.ram	body	integer	The amount of RAM a flavor has, in MiB.**New in version 2.47**
flavor.disk	body	integer	The size of the root disk that was created in GiB.**New in version 2.47**
flavor.ephemeral	body	integer	The size of the ephemeral disk that was created, in GiB.**New in version 2.47**
flavor.swap	body	integer	The size of a dedicated swap disk that was allocated, in MiB.**New in version 2.47**
flavor.original_name	body	string	The display name of a flavor.**New in version 2.47**
flavor.extra_specs (Optional)	body	object	A dictionary of the flavor's extra-specs key-and-value pairs. This will only be included if the user is allowed by policy to index flavor extra_specs.**New in version 2.47**
flavor.extra_specs.key	body	string	The extra spec key of a flavor.**New in version 2.47**
flavor.extra_specs.value	body	string	The extra spec value of a flavor.**New in version 2.47**
hostId	body	string	An ID string representing the host. This is a hashed value so will not actually look like a hostname, and is hashed with data from the project_id, so the same physical host as seen by two different project_ids, will be different. It is useful when within the same project you need to determine if two instances are on the same or different physical hosts for the purposes of availability or performance.
id	body	string	The UUID of the server.
image	body	object	The UUID and links for the image for your server instance. The `image` object might be an empty string when you boot the server from a volume.
key_name	body	string	The name of associated key pair, if any.
links	body	array	Links to the resources in question. See API Guide / Links and References for more info.
metadata	body	object	A dictionary of metadata key-and-value pairs, which is maintained for backward compatibility.
name	body	string	The server name.
OS-DCF:diskConfig	body	string	Disk configuration. The value is either: `AUTO`. The API builds the server with a single partition the size of the target flavor disk. The API automatically adjusts the file system to fit the entire partition. `MANUAL`. The API builds the server by using the partition scheme and file system that is in the source image. If the target flavor disk is larger, The API does not partition the remaining disk space.
OS-EXT-AZ:availability_zone	body	string	The availability zone name.
OS-EXT-SRV-ATTR:host	body	string	The name of the compute host on which this instance is running. Appears in the response for administrative users only.
OS-EXT-SRV-ATTR:hypervisor_hostname	body	string	The hypervisor host name provided by the Nova virt driver. For the Ironic driver, it is the Ironic node uuid. Appears in the response for administrative users only.
OS-EXT-SRV-ATTR:instance_name	body	string	The instance name. The Compute API generates the instance name from the instance name template. Appears in the response for administrative users only.
OS-EXT-STS:power_state	body	integer	The power state of the instance. This is an enum value that is mapped as: `0: NOSTATE 1: RUNNING 3: PAUSED 4: SHUTDOWN 6: CRASHED 7: SUSPENDED`
OS-EXT-STS:task_state	body	string	The task state of the instance.
OS-EXT-STS:vm_state	body	string	The VM state.
os-extended-volumes:volumes_attached	body	array	The attached volumes, if any.
os-extended-volumes:volumes_attached.id	body	string	The attached volume ID.
os-extended-volumes:volumes_attached.delete_on_termination	body	boolean	A flag indicating if the attached volume will be deleted when the server is deleted. By default this is False and can only be set when creating a volume while creating a server, which is commonly referred to as boot from volume.**New in version 2.3**
OS-SRV-USG:launched_at	body	string	The date and time when the server was launched.The date and time stamp format is ISO 8601: `CCYY-MM-DDThh:mm:ss±hh:mm` For example, `2015-08-27T09:49:58-05:00`.The `hh±:mm` value, if included, is the time zone as an offset from UTC. If the `deleted_at` date and time stamp is not set, its value is `null`.
OS-SRV-USG:terminated_at	body	string	The date and time when the server was deleted.The date and time stamp format is ISO 8601: `CCYY-MM-DDThh:mm:ss±hh:mm` For example, `2015-08-27T09:49:58-05:00`. The `±hh:mm` value, if included, is the time zone as an offset from UTC. If the `deleted_at` date and time stamp is not set, its value is `null`.
security_groups	body	array	One or more security groups objects.
security_group.name	body	string	The security group name.
status	body	string	The server status.
tenant_id	body	string	The UUID of the tenant in a multi-tenancy cloud.
updated	body	string	The date and time when the resource was updated. The date and time stamp format is ISO 8601 `CCYY-MM-DDThh:mm:ss±hh:mm` For example, `2015-08-27T09:49:58-05:00`. The `±hh:mm` value, if included, is the time zone as an offset from UTC. In the previous example, the offset value is `-05:00`.
user_id	body	string	The user ID of the user who owns the server.
fault (Optional)	body	object	A fault object. Only displayed when the server status is `ERROR` or `DELETED` and a fault occurred.
fault.code	body	integer	The error response code.

fault.created	body	string	The date and time when the exception was raised. The date and time stamp format is ISO 8601 `CCYY-MM-DDThh:mm:ss±hh:mm` For example, `2015-08-27T09:49:58-05:00`. The `±hh:mm` value, if included, is the time zone as an offset from UTC. In the previous example, the offset value is `-05:00`.
fault.message	body	string	The error message.
fault.details (Optional)	body	string	The stack trace. It is available if the response code is not 500 or you have the administrator privilege
progress (Optional)	body	integer	A percentage value of the operation progress. This parameter only appears when the server status is `ACTIVE`, `BUILD`, `REBUILD`, `RESIZE`, `VERIFY_RESIZE` or `MIGRATING`.
OS-EXT-SRV-ATTR:hostname (Optional)	body	string	The hostname set on the instance when it is booted. By default, it appears in the response for administrative users only.**New in version 2.3**
OS-EXT-SRV-ATTR:reservation_id (Optional)	body	string	The reservation id for the server. This is an id that can be useful in tracking groups of servers created with multiple create, that will all have the same reservation_id. By default, it appears in the response for administrative users only.**New in version 2.3**
OS-EXT-SRV-ATTR:launch_index (Optional)	body	integer	When servers are launched via multiple create, this is the sequence in which the servers were launched. By default, it appears in the response for administrative users only.**New in version 2.3**
OS-EXT-SRV-ATTR:kernel_id (Optional)	body	string	The UUID of the kernel image when using an AMI. Will be null if not. By default, it appears in the response for administrative users only.**New in version 2.3**
OS-EXT-SRV-ATTR:ramdisk_id (Optional)	body	string	The UUID of the ramdisk image when using an AMI. Will be null if not. By default, it appears in the response for administrative users only.**New in version 2.3**
OS-EXT-SRV-ATTR:root_device_name (Optional)	body	string	The root device name for the instance By default, it appears in the response for administrative users only.**New in version 2.3**
OS-EXT-SRV-ATTR:user_data (Optional)	body	string	The user_data the instance was created with. By default, it appears in the response for administrative users only.**New in version 2.3**
locked	body	boolean	True if the instance is locked otherwise False.**New in version 2.9**
host_status (Optional)	body	string	The host status. Values where next value in list can override the previous: `UP` if nova-compute up. `UNKNOWN` if nova-compute not reported by servicegroup driver. `DOWN` if nova-compute forced down. `MAINTENANCE` if nova-compute is disabled.Empty string indicates there is no host for server.This attribute appears in the response only if the policy permits. By default, only administrators can get this parameter.**New in version 2.16**
description	body	string	The description of the server. Before microversion 2.19 this was set to the server name.**New in version 2.19**
tags	body	array	A list of tags. The maximum count of tags in this list is 50.**New in version 2.26**
trusted_image_certificates	body	array	A list of trusted certificate IDs, that were used during image signature verification to verify the signing certificate. The list is restricted to a maximum of 50 IDs. The value is `null` if trusted certificate IDs are not set.**New in version 2.63**

Example Show Server Details (2.63)


```
{
    "server": {
        "OS-DCF:diskConfig": "AUTO",
        "OS-EXT-AZ:availability_zone": "nova",
        "OS-EXT-SRV-ATTR:host": "compute",
        "OS-EXT-SRV-ATTR:hostname": "new-server-test",
        "OS-EXT-SRV-ATTR:hypervisor_hostname": "fake-mini",
        "OS-EXT-SRV-ATTR:instance_name": "instance-00000001",
        "OS-EXT-SRV-ATTR:kernel_id": "",
        "OS-EXT-SRV-ATTR:launch_index": 0,
        "OS-EXT-SRV-ATTR:ramdisk_id": "",
        "OS-EXT-SRV-ATTR:reservation_id": "r-ov3q80zj",
        "OS-EXT-SRV-ATTR:root_device_name": "/dev/sda",
        "OS-EXT-SRV-ATTR:user_data": "IyEvYmluL2Jhc2gKL2Jpbi9zdQplY2hvICJJIGFtIGluIHlvdSEiCg==",
        "OS-EXT-STS:power_state": 1,
        "OS-EXT-STS:task_state": null,
        "OS-EXT-STS:vm_state": "active",
        "OS-SRV-USG:launched_at": "2017-02-14T19:23:59.895661",
        "OS-SRV-USG:terminated_at": null,
        "accessIPv4": "1.2.3.4",
        "accessIPv6": "80fe::",
        "addresses": {
            "private": [
                {
                    "OS-EXT-IPS-MAC:mac_addr": "aa:bb:cc:dd:ee:ff",
                    "OS-EXT-IPS:type": "fixed",
                    "addr": "192.168.0.3",
                    "version": 4
                }
            ]
        },
        "config_drive": "",
        "created": "2017-02-14T19:23:58Z",
        "description": null,
        "flavor": {
            "disk": 1,
            "ephemeral": 0,
            "extra_specs": {
                "hw:cpu_policy": "dedicated",
                "hw:mem_page_size": "2048"
            },
            "original_name": "m1.tiny.specs",
            "ram": 512,
            "swap": 0,
```

```
            "vcpus": 1
        },
        "hostId": "2091634baaccdc4c5a1d57069c833e402921df696b7f970791b12ec6",
        "host_status": "UP",
        "id": "9168b536-cd40-4630-b43f-b259807c6e87",
        "image": {
            "id": "70a599e0-31e7-49b7-b260-868f441e862b",
            "links": [
                {
                    "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/70a599e0-
31e7-49b7-b260-868f441e862b",
                    "rel": "bookmark"
                }
            ]
        },
        "key_name": null,
        "links": [
            {
                "href":
"http://openstack.example.com/v2.1/6f70656e737461636b20342065766572/servers/9168b536-cd40-4630-b43f-
b259807c6e87",
                "rel": "self"
            },
            {
                "href": "http://openstack.example.com/6f70656e737461636b20342065766572/servers/9168b536-
cd40-4630-b43f-b259807c6e87",
                "rel": "bookmark"
            }
        ],
        "locked": false,
        "metadata": {
            "My Server Name": "Apache1"
        },
        "name": "new-server-test",
        "os-extended-volumes:volumes_attached": [],
        "progress": 0,
        "security_groups": [
            {
                "name": "default"
            }
        ],
        "status": "ACTIVE",
        "tags": [],
        "tenant_id": "6f70656e737461636b20342065766572",
        "trusted_image_certificates": [
            "0b5d2c72-12cc-4ba6-a8d7-3ff5cc1d8cb8",
            "674736e3-f25c-405c-8362-bbf991e0ce0a"
        ],
        "updated": "2017-02-14T19:24:00Z",
        "user_id": "fake"
    }
}
```


Delete Server DELETE /servers/{server_id}

Deletes a server.

By default, the instance is going to be (hard) deleted immediately from the system, but you can set `reclaim_instance_interval` > 0 to make the API soft delete the instance, so that the instance won't be deleted until the `reclaim_instance_interval` has expired since the instance was soft deleted. The instance marked as `SOFT_DELETED` can be recovered via `restore` action before it's really deleted from the system.

Preconditions

- The server must exist.
- Anyone can delete a server when the status of the server is not locked and when the policy allows.
- If the server is locked, you must have administrator privileges to delete the server.

Asynchronous postconditions

- With correct permissions, you can see the server status as `deleting`.
- The ports attached to the server, which Nova created during the server create process or when attaching interfaces later, are deleted.
- The server does not appear in the list servers response.
- If hard delete, the server managed by OpenStack Compute is deleted on the compute node.

Troubleshooting

- If server status remains in `deleting` status or another error status, the request failed. Ensure that you meet the preconditions. Then, investigate the compute back end.

- The request returns the HTTP 409 response code when the server is locked even if you have correct permissions. Ensure that you meet the preconditions then investigate the server status.
- The server managed by OpenStack Compute is not deleted from the compute node.

Normal response codes: 204

Error response codes: unauthorized(401), forbidden(403), itemNotFound(404), conflict(409)

Request¶

Name	In	Type	Description
server_id	path	string	The UUID of the server.

Response¶

There is no body content for the response of a successful DELETE query

Hypervisors (os-hypervisors)¶

Lists all hypervisors, shows summary statistics for all hypervisors over all compute nodes, shows details for a hypervisor, shows the uptime for a hypervisor, lists all servers on hypervisors that match the given `hypervisor_hostname_pattern` or searches for hypervisors by the given `hypervisor_hostname_pattern`.

List Hypervisors GET /os-hypervisors

Lists hypervisors.

Policy defaults enable only users with the administrative role to perform this operation. Cloud providers can change these permissions through the `policy.json` file.

Normal response codes: 200

Error response codes: badRequest(400), unauthorized(401), forbidden(403)

Request¶

Name	In	Type	Description
limit (Optional)	query	integer	Requests a page size of items. Returns a number of items up to a limit value. Use the `limit` parameter to make an initial limited request and use the ID of the last-seen item from the response as the `marker` parameter value in a subsequent limited request.**New in version 2.33**
marker (Optional)	query	integer	The ID of the last-seen item. Use the `limit` parameter to make an initial limited request and use the ID of the last-seen item from the response as the `marker` parameter value in a subsequent limited request.**New in version 2.33****Available until version 2.52**
marker (Optional)	query	string	The ID of the last-seen item as a UUID. Use the `limit` parameter to make an initial limited request and use the ID of the last-seen item from the response as the `marker` parameter value in a subsequent limited request.**New in version 2.53**
hypervisor_hostname_pattern (Optional)	query	string	The hypervisor host name or a portion of it. The hypervisor hosts are selected with the host name matching this pattern. Note `limit` and `marker` query parameters for paging are not supported when listing hypervisors using a hostname pattern. Also, `links` will not be returned in the response when using this query parameter.**New in version 2.53**
with_servers (Optional)	query	boolean	Include all servers which belong to each hypervisor in the response output.**New in version 2.53**

Response¶

Name	In	Type	Description
hypervisors	body	array	An array of hypervisor information.
hypervisor_hostname	body	string	The hypervisor host name provided by the Nova virt driver. For the Ironic driver, it is the Ironic node uuid.
id	body	integer	The id of the hypervisor.**Available until version 2.52**
id	body	string	The id of the hypervisor as a UUID.**New in version 2.53**
state	body	string	The state of the hypervisor. One of `up` or `down`.
status	body	string	The status of the hypervisor. One of `enabled` or `disabled`.
hypervisor_links (Optional)	body	array	Links to the hypervisors resource. See API Guide / Links and References for more info.**New in version 2.33**
servers (Optional)	body	array	A list of `server` objects.**New in version 2.53**
servers.uuid (Optional)	body	string	The server ID.**New in version 2.53**
servers.name (Optional)	body	string	The server name.**New in version 2.53**

Example List Hypervisors (v2.33): JSON response

```
{
    "hypervisors": [
        {
            "hypervisor_hostname": "host1",
            "id": 2,
            "state": "up",
            "status": "enabled"
        }
    ],
    "hypervisors_links": [
        {
            "href": "http://openstack.example.com/v2.1/6f70656e737461636b20342065766572/hypervisors?
limit=1&marker=2",
            "rel": "next"
        }
    ]
}
```

Example List Hypervisors With Servers (v2.53): JSON response

```
{
    "hypervisors": [
        {
            "hypervisor_hostname": "fake-mini",
            "id": "b1e43b5f-eec1-44e0-9f10-7b4945c0226d",
            "state": "up",
            "status": "enabled",
            "servers": [
                {
                    "name": "test_server1",
                    "uuid": "aaaaaaaa-aaaa-aaaa-aaaa-aaaaaaaaaaaa"
                },
                {
                    "name": "test_server2",
                    "uuid": "bbbbbbbb-bbbb-bbbb-bbbb-bbbbbbbbbbbb"
                }
            ]
        }
    ]
}
```

List Hypervisors Details GET /os-hypervisors/detail

Lists hypervisors details.

Policy defaults enable only users with the administrative role to perform this operation. Cloud providers can change these permissions through the `policy.json` file.

Normal response codes: 200

Error response codes: badRequest(400), unauthorized(401), forbidden(403)

Request¶

Name	In	Type	Description
limit (Optional)	query	integer	Requests a page size of items. Returns a number of items up to a limit value. Use the `limit` parameter to make an initial limited request and use the ID of the last-seen item from the response as the `marker` parameter value in a subsequent limited request.**New in version 2.33**
marker (Optional)	query	integer	The ID of the last-seen item. Use the `limit` parameter to make an initial limited request and use the ID of the last-seen item from the response as the `marker` parameter value in a subsequent limited request.**New in version 2.33****Available until version 2.52**
marker (Optional)	query	string	The ID of the last-seen item as a UUID. Use the `limit` parameter to make an initial limited request and use the ID of the last-seen item from the response as the `marker` parameter value in a subsequent limited request.**New in version 2.53**
hypervisor_hostname_pattern (Optional)	query	string	The hypervisor host name or a portion of it. The hypervisor hosts are selected with the host name matching this pattern. Note `limit` and `marker` query parameters for paging are not supported when listing hypervisors using a hostname pattern. Also, `links` will not be returned in the response when using this query parameter.**New in version 2.53**
with_servers (Optional)	query	boolean	Include all servers which belong to each hypervisor in the response output.**New in version 2.53**

Response¶

Name	In	Type	Description
hypervisors	body	array	An array of hypervisor information.
cpu_info	body	object	A dictionary that contains cpu information like `arch`, `model`, `vendor`, `features` and `topology`. The content of this field is hypervisor specific. NoteSince version 2.28 `cpu_info` field is returned as a dictionary instead of string.
current_workload	body	integer	The current_workload is the number of tasks the hypervisor is responsible for. This will be equal or greater than the number of active VMs on the system (it can be greater when VMs are being deleted and the hypervisor is still cleaning up).
status	body	string	The status of the hypervisor. One of `enabled` or `disabled`.
state	body	string	The state of the hypervisor. One of `up` or `down`.
disk_available_least	body	integer	The actual free disk on this hypervisor(in GiB).
host_ip	body	string	The IP address of the hypervisor's host.
free_disk_gb	body	integer	The free disk remaining on this hypervisor(in GiB).
free_ram_mb	body	integer	The free RAM in this hypervisor(in MiB).
hypervisor_hostname	body	string	The hypervisor host name provided by the Nova virt driver. For the Ironic driver, it is the Ironic node uuid.
hypervisor_type	body	string	The hypervisor type.
hypervisor_version	body	integer	The hypervisor version.
id	body	integer	The id of the hypervisor.**Available until version 2.52**
id	body	string	The id of the hypervisor as a UUID.**New in version 2.53**
local_gb	body	integer	The disk in this hypervisor(in GiB).
local_gb_used	body	integer	The disk used in this hypervisor(in GiB).
memory_mb	body	integer	The memory of this hypervisor(in MiB).
memory_mb_used	body	integer	The memory used in this hypervisor(in MiB).
running_vms	body	integer	The number of running vms on this hypervisor.
servers (Optional)	body	array	A list of `server` objects.**New in version 2.53**
servers.uuid (Optional)	body	string	The server ID.**New in version 2.53**
servers.name (Optional)	body	string	The server name.**New in version 2.53**
service	body	object	The hypervisor service object.
service.host	body	string	The name of the host.
service.id	body	integer	The id of the service.**Available until version 2.52**
service.id	body	string	The id of the service as a uuid.**New in version 2.53**
service.disable_reason	body	string	The disable reason of the service, `null` if the service is enabled or disabled without reason provided.
vcpus	body	integer	The number of vcpu in this hypervisor.
vcpus_used	body	integer	The number of vcpu used in this hypervisor.
hypervisor_links (Optional)	body	array	Links to the hypervisors resource. See API Guide / Links and References for more info.**New in version 2.33**

Example List Hypervisors Details (v2.33): JSON response

```
{
    "hypervisors": [
        {
            "cpu_info": {
                "arch": "x86_64",
                "model": "Nehalem",
                "vendor": "Intel",
                "features": [
                    "pge",
                    "clflush"
                ],
                "topology": {
                    "cores": 1,
                    "threads": 1,
                    "sockets": 4
                }
```

```
            },
            "current_workload": 0,
            "status": "enabled",
            "state": "up",
            "disk_available_least": 0,
            "host_ip": "1.1.1.1",
            "free_disk_gb": 1028,
            "free_ram_mb": 7680,
            "hypervisor_hostname": "host1",
            "hypervisor_type": "fake",
            "hypervisor_version": 1000,
            "id": 2,
            "local_gb": 1028,
            "local_gb_used": 0,
            "memory_mb": 8192,
            "memory_mb_used": 512,
            "running_vms": 0,
            "service": {
                "host": "host1",
                "id": 7,
                "disabled_reason": null
            },
            "vcpus": 2,
            "vcpus_used": 0
        }
    ],
    "hypervisors_links": [
        {
            "href": "http://openstack.example.com/v2.1/6f70656e737461636b20342065766572/hypervisors/detail?
limit=1&marker=2",
            "rel": "next"
        }
    ]
}
```


Example List Hypervisors Details (v2.53): JSON response


```
{
    "hypervisors": [
        {
            "cpu_info": {
                "arch": "x86_64",
                "model": "Nehalem",
                "vendor": "Intel",
                "features": [
                    "pge",
                    "clflush"
                ],
                "topology": {
                    "cores": 1,
                    "threads": 1,
                    "sockets": 4
                }
            },
            "current_workload": 0,
            "status": "enabled",
            "state": "up",
            "disk_available_least": 0,
            "host_ip": "1.1.1.1",
            "free_disk_gb": 1028,
            "free_ram_mb": 7680,
            "hypervisor_hostname": "host2",
            "hypervisor_type": "fake",
            "hypervisor_version": 1000,
            "id": "1bb62a04-c576-402c-8147-9e89757a09e3",
            "local_gb": 1028,
            "local_gb_used": 0,
            "memory_mb": 8192,
            "memory_mb_used": 512,
            "running_vms": 0,
            "service": {
                "host": "host1",
                "id": "62f62f6e-a713-4cbe-87d3-3ecf8a1e0f8d",
                "disabled_reason": null
            },
            "vcpus": 2,
            "vcpus_used": 0
        }
    ],
```

```
    "hypervisors_links": [
        {
            "href": "http://openstack.example.com/v2.1/6f70656e737461636b20342065766572/hypervisors/detail?
limit=1&marker=1bb62a04-c576-402c-8147-9e89757a09e3",
            "rel": "next"
        }
    ]
}
```

Show Hypervisor Statistics GET /os-hypervisors/statistics

Shows summary statistics for all enabled hypervisors over all compute nodes.

Policy defaults enable only users with the administrative role to perform this operation. Cloud providers can change these permissions through the `policy.json` file.

Normal response codes: 200

Error response codes: unauthorized(401), forbidden(403)

Response¶

Name	In	Type	Description
hypervisor_statistics	body	object	The hypervisors statistics summary object.
count	body	integer	The number of hypervisors.
current_workload	body	integer	The current_workload is the number of tasks the hypervisor is responsible for. This will be equal or greater than the number of active VMs on the system (it can be greater when VMs are being deleted and the hypervisor is still cleaning up).
disk_available_least	body	integer	The actual free disk on all hypervisors(in GiB).
free_disk_gb	body	integer	The free disk remaining on all hypervisors(in GiB).
free_ram_mb	body	integer	The free RAM on all hypervisors(in MiB).
local_gb	body	integer	The disk on all hypervisors(in GiB).
local_gb_used	body	integer	The disk used on all hypervisors(in GiB).
memory_mb	body	integer	The memory of all hypervisors(in MiB).
memory_mb_used	body	integer	The memory used on all hypervisors(in MiB).
running_vms	body	integer	The total number of running vms on all hypervisors.
vcpus	body	integer	The number of vcpu on all hypervisors.
vcpus_used	body	integer	The number of vcpu used on all hypervisors.

Example Show Hypervisor Statistics: JSON response

```
{
    "hypervisor_statistics": {
        "count": 1,
        "current_workload": 0,
        "disk_available_least": 0,
        "free_disk_gb": 1028,
        "free_ram_mb": 7680,
        "local_gb": 1028,
        "local_gb_used": 0,
        "memory_mb": 8192,
        "memory_mb_used": 512,
        "running_vms": 0,
        "vcpus": 2,
        "vcpus_used": 0
    }
}
```

Show Hypervisor Details GET /os-hypervisors/{hypervisor_id

Servers with volume attachments (servers, os-volume_attachments)¶

Attaches volumes that are created through the volume API to server instances. Also, lists volume attachments for a server, shows details for a volume attachment, and detaches a volume.

List volume attachments for an instance GET /servers/{server_id}/os-volume_attachments

List volume attachments for an instance.

Normal response codes: 200

Error response codes: unauthorized(401), forbidden(403), itemNotFound(404)

Request¶

Name	In	Type	Description
server_id	path	string	The UUID of the server.
limit (Optional)	query	integer	Used in conjunction with `offset` to return a slice of items. `limit` is the maximum number of items to return. If `limit` is not specified, or exceeds the configurable `max_limit`, then `max_limit` will be used instead.
offset (Optional)	query	integer	Used in conjunction with `limit` to return a slice of items. `offset` is where to start in the list.

Response¶

Name	In	Type	Description
volumeAttachments	body	array	The list of volume attachments.
device	body	string	Name of the device such as, `/dev/vdb`.
id	body	string	The UUID of the attachment.
serverId	body	string	The UUID of the server.
volumeId	body	string	The UUID of the attached volume.

Example List volume attachments for an instance: JSON response

```
{
    "volumeAttachments": [
        {
            "device": "/dev/sdd",
            "id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
            "serverId": "4d8c3732-a248-40ed-bebc-539a6ffd25c0",
            "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f803"
        },
        {
            "device": "/dev/sdc",
            "id": "a26887c6-c47b-4654-abb5-dfadf7d3f804",
            "serverId": "4d8c3732-a248-40ed-bebc-539a6ffd25c0",
            "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f804"
        }
    ]
}
```

Attach a volume to an instance POST /servers/{server_id}/os-volume_attachments

Attach a volume to an instance.

Normal response codes: 200

Error response codes: badRequest(400), unauthorized(401), forbidden(403), itemNotFound(404), conflict(409)

> Note
>
> From v2.20 attach a volume to an instance in SHELVED or SHELVED_OFFLOADED state is allowed.

> Note
>
> From v2.60, attaching a multiattach volume to multiple instances is supported for instances that are not SHELVED_OFFLOADED. The ability to actually support a multiattach volume depends on the volume type and compute hosting the instance.

Request¶

Name	In	Type	Description
server_id	path	string	The UUID of the server.
volumeAttachment	body	object	A dictionary representation of a volume attachment containing the fields `device` and `volumeId`.
volumeId	body	string	The UUID of the volume to attach.
device (Optional)	body	string	Name of the device such as, `/dev/vdb`. Omit or set this parameter to null for auto-assignment, if supported. If you specify this parameter, the device must not exist in the guest operating system. Note that as of the 12.0.0 Liberty release, the Nova libvirt driver no longer honors a user-supplied device name. This is the same behavior as if the device name parameter is not supplied on the request.
tag (Optional)	body	string	A device role tag that can be applied to a volume when attaching it to the VM. The guest OS of a server that has devices tagged in this manner can access hardware metadata about the tagged devices from the metadata API and on the config drive, if enabled. NoteTagged volume attachment is not supported for shelved-offloaded instances.**New in version 2.49**

Example Attach a volume to an instance: JSON request

```
{
    "volumeAttachment": {
        "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
        "device": "/dev/vdd"
    }
}
```

Example Attach a volume to an instance and tag it (v2.49): JSON request

```
{
    "volumeAttachment": {
        "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
        "tag": "foo"
    }
}
```

Response¶

Name	In	Type	Description
volumeAttachment	body	object	A dictionary representation of a volume attachment containing the fields `device`, `id`, `serverId` and `volumeId`.
device	body	string	Name of the device such as, `/dev/vdb`.
id	body	string	The UUID of the attachment.
serverId	body	string	The UUID of the server.
volumeId	body	string	The UUID of the attached volume.

Example Attach a volume to an instance: JSON response

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

Detach a volume from an instance DELETE /servers/{server_id}/os-volume_attachments/{volume_id}

Detach a volume from an instance.

Normal response codes: 202

Error response codes: badRequest(400), unauthorized(401), forbidden(403), itemNotFound(404), conflict(409)

> Note
>
> From v2.20 detach a volume from an instance in SHELVED or SHELVED_OFFLOADED state is allowed.

Request¶

Name	In	Type	Description
server_id	path	string	The UUID of the server.
volume_id	path	string	The UUID of the volume to detach.

Response¶

No body is returned on successful request.

Flavors¶

Show and manage server flavors.

Flavors are a way to describe the basic dimensions of a server to be created including how much `cpu`, `ram`, and `disk space` are allocated to a server built with this flavor.

List Flavors GET /flavors

Lists all flavors accessible to your project.

Normal response codes: 200

Error response codes: unauthorized(401), forbidden(403)

Request¶

Name	In	Type	Description
sort_key (Optional)	query	string	Sorts by a flavor attribute. Default attribute is `flavorid`. You can specify multiple pairs of sort key and sort direction query parameters. If you omit the sort direction in a pair, the API uses the natural sorting direction of the flavor `sort_key` attribute.
sort_dir (Optional)	query	string	Sort direction. A valid value is `asc` (ascending) or `desc` (descending). Default is `asc`. You can specify multiple pairs of sort key and sort direction query parameters. If you omit the sort direction in a pair, the API uses the natural sorting direction of the direction of the flavor `sort_key` attribute.
limit (Optional)	query	integer	Requests a page size of items. Returns a number of items up to a limit value. Use the `limit` parameter to make an initial limited request and use the ID of the last-seen item from the response as the `marker` parameter value in a subsequent limited request.
marker (Optional)	query	string	The ID of the last-seen item. Use the `limit` parameter to make an initial limited request and use the ID of the last-seen item from the response as the `marker` parameter value in a subsequent limited request.
minDisk (Optional)	query	integer	Filters the response by a minimum disk space, in GiB. For example, `100`.
minRam (Optional)	query	integer	Filters the response by a minimum RAM, in MiB. For example, `512`.
is_public (Optional)	query	string	This parameter is only applicable to users with the administrative role. For all other non-admin users, the parameter is ignored and only public flavors will be returned. Filters the flavor list based on whether the flavor is public or private. If the value of this parameter is not specified, it is treated as `True`. If the value is specified, `1`, `t`, `true`, `on`, `y` and `yes` are treated as `True`. `0`, `f`, `false`, `off`, `n` and `no` are treated as `False` (they are case-insensitive). If the value is `None` (case-insensitive) both public and private flavors will be listed in a single request.

Response¶

Name	In	Type	Description
flavors	body	array	An array of flavor objects.
id	body	string	The ID of the flavor. While people often make this look like an int, this is really a string.
name	body	string	The display name of a flavor.
description	body	string	The description of the flavor.**New in version 2.55**
links	body	array	Links to the resources in question. See API Guide / Links and Referencesfor more info.

Example List Flavors (v2.55)


```
{
    "flavors": [
        {
            "id": "1",
            "links": [
                {
                    "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/flavors/1",
                    "rel": "self"
                },
                {
                    "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/1",
                    "rel": "bookmark"
                }
            ],
            "name": "m1.tiny",
            "description": null
        },
        {
            "id": "2",
            "links": [
                {
                    "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/flavors/2",
                    "rel": "self"
                },
                {
                    "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/2",
                    "rel": "bookmark"
                }
            ],
            "name": "m1.small",
            "description": null
        },
        {
            "id": "3",
            "links": [
                {
                    "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/flavors/3",
                    "rel": "self"
                },
                {
                    "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/3",
                    "rel": "bookmark"
                }
            ],
            "name": "m1.medium",
            "description": null
        },
        {
            "id": "4",
            "links": [
                {
                    "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/flavors/4",
                    "rel": "self"
                },
                {
                    "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/4",
                    "rel": "bookmark"
                }
            ],
            "name": "m1.large",
            "description": null
        },
        {
            "id": "5",
            "links": [
                {
                    "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/flavors/5",
                    "rel": "self"
                },
                {
                    "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/5",
                    "rel": "bookmark"
                }
            ],
            "name": "m1.xlarge",
            "description": null
        },
```

```
        {
            "id": "6",
            "links": [
                {
                    "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/flavors/6",
                    "rel": "self"
                },
                {
                    "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/6",
                    "rel": "bookmark"
                }
            ],
            "name": "m1.tiny.specs",
            "description": null
        },
        {
            "id": "7",
            "links": [
                {
                    "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/flavors/7",
                    "rel": "self"
                },
                {
                    "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/7",
                    "rel": "bookmark"
                }
            ],
            "name": "m1.small.description",
            "description": "test description"
        }
    ]
}
```


Create Flavor POST /flavors

Creates a flavor.

Creating a flavor is typically only available to administrators of a cloud because this has implications for scheduling efficiently in the cloud.

Normal response codes: 200

Error response codes: badRequest(400), unauthorized(401), forbidden(403), conflict(409)

Request¶

Name	In	Type	Description
flavor	body	object	The ID and links for the flavor for your server instance. A flavor is a combination of memory, disk size, and CPUs.
name	body	string	The display name of a flavor.
description (Optional)	body	string	A free form description of the flavor. Limited to 65535 characters in length. Only printable characters are allowed.**New in version 2.55**
id (Optional)	body	string	The ID of the flavor. While people often make this look like an int, this is really a string. If not provided, this defaults to a uuid.
ram	body	integer	The amount of RAM a flavor has, in MiB.
disk	body	integer	The size of the root disk that will be created in GiB. If 0 the root disk will be set to exactly the size of the image used to deploy the instance. However, in this case filter scheduler cannot select the compute host based on the virtual image size. Therefore, 0 should only be used for volume booted instances or for testing purposes. Volume-backed instances can be enforced for flavors with zero root disk via the `os_compute_api:servers:create:zero_disk_flavor` policy rule.
vcpus	body	integer	The number of virtual CPUs that will be allocated to the server.
OS-FLV-EXT-DATA:ephemeral (Optional)	body	integer	The size of the ephemeral disk that will be created, in GiB. Ephemeral disks may be written over on server state changes. So should only be used as a scratch space for applications that are aware of its limitations. Defaults to 0.
swap (Optional)	body	integer	The size of a dedicated swap disk that will be allocated, in MiB. If 0 (the default), no dedicated swap disk will be created.
rxtx_factor (Optional)	body	float	The receive / transmit factor (as a float) that will be set on ports if the network backend supports the QOS extension. Otherwise it will be ignored. It defaults to 1.0.
os-flavor-access:is_public (Optional)	body	boolean	Whether the flavor is public (available to all projects) or scoped to a set of projects. Default is True if not specified.

Example Create Flavor (v2.55)

```
{
    "flavor": {
        "name": "test_flavor",
        "ram": 1024,
        "vcpus": 2,
        "disk": 10,
        "id": "10",
        "rxtx_factor": 2.0,
        "description": "test description"
    }
}
```

Response¶

Name	In	Type	Description
flavor	body	object	The ID and links for the flavor for your server instance. A flavor is a combination of memory, disk size, and CPUs.
name	body	string	The display name of a flavor.
description	body	string	The description of the flavor.**New in version 2.55**
id	body	string	The ID of the flavor. While people often make this look like an int, this is really a string.
ram	body	integer	The amount of RAM a flavor has, in MiB.
disk	body	integer	The size of the root disk that will be created in GiB. If 0 the root disk will be set to exactly the size of the image used to deploy the instance. However, in this case filter scheduler cannot select the compute host based on the virtual image size. Therefore, 0 should only be used for volume booted instances or for testing purposes. Volume-backed instances can be enforced for flavors with zero root disk via the `os_compute_api:servers:create:zero_disk_flavor` policy rule.
vcpus	body	integer	The number of virtual CPUs that will be allocated to the server.
links	body	array	Links to the resources in question. See API Guide / Links and Referencesfor more info.
OS-FLV-EXT-DATA:ephemeral	body	integer	The size of the ephemeral disk that will be created, in GiB. Ephemeral disks may be written over on server state changes. So should only be used as a scratch space for applications that are aware of its limitations. Defaults to 0.
OS-FLV-DISABLED:disabled (Optional)	body	boolean	Whether or not the flavor has been administratively disabled. This is typically only visible to administrative users.
swap	body	integer	The size of a dedicated swap disk that will be allocated, in MiB. If 0 (the default), no dedicated swap disk will be created. Currently, the empty string ('') is used to represent 0.
rxtx_factor	body	float	The receive / transmit factor (as a float) that will be set on ports if the network backend supports the QOS extension. Otherwise it will be ignored. It defaults to 1.0.
os-flavor-access:is_public	body	boolean	Whether the flavor is public (available to all projects) or scoped to a set of projects. Default is True if not specified.
extra_specs (Optional)	body	object	A dictionary of the flavor's extra-specs key-and-value pairs. This will only be included if the user is allowed by policy to index flavor extra_specs.**New in version 2.61**

Example Create Flavor (v2.61)

```
{
    "flavor": {
        "OS-FLV-DISABLED:disabled": false,
        "disk": 10,
        "OS-FLV-EXT-DATA:ephemeral": 0,
        "os-flavor-access:is_public": true,
        "id": "10",
        "links": [
            {
                "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/flavors/10",
                "rel": "self"
            },
            {
                "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/10",
                "rel": "bookmark"
            }
        ],
        "name": "test_flavor",
        "ram": 1024,
        "swap": "",
        "rxtx_factor": 2.0,
        "vcpus": 2,
        "description": "test description",
        "extra_specs": {}
    }
}
```

List Flavors With Details GET /flavors/detail

Lists flavors with details.

Normal response codes: 200

Error response codes: badRequest(400), unauthorized(401), forbidden(403)

Request¶

Name	In	Type	Description
sort_key (Optional)	query	string	Sorts by a flavor attribute. Default attribute is `flavorid`. You can specify multiple pairs of sort key and sort direction query parameters. If you omit the sort direction in a pair, the API uses the natural sorting direction of the flavor `sort_key` attribute.
sort_dir (Optional)	query	string	Sort direction. A valid value is `asc` (ascending) or `desc` (descending). Default is `asc`. You can specify multiple pairs of sort key and sort direction query parameters. If you omit the sort direction in a pair, the API uses the natural sorting direction of the direction of the flavor `sort_key` attribute.
limit (Optional)	query	integer	Requests a page size of items. Returns a number of items up to a limit value. Use the `limit` parameter to make an initial limited request and use the ID of the last-seen item from the response as the `marker` parameter value in a subsequent limited request.
marker (Optional)	query	string	The ID of the last-seen item. Use the `limit` parameter to make an initial limited request and use the ID of the last-seen item from the response as the `marker` parameter value in a subsequent limited request.
minDisk (Optional)	query	integer	Filters the response by a minimum disk space, in GiB. For example, `100`.
minRam (Optional)	query	integer	Filters the response by a minimum RAM, in MiB. For example, `512`.
is_public (Optional)	query	string	This parameter is only applicable to users with the administrative role. For all other non-admin users, the parameter is ignored and only public flavors will be returned. Filters the flavor list based on whether the flavor is public or private. If the value of this parameter is not specified, it is treated as `True`. If the value is specified, `1`, `t`, `true`, `on`, `y` and `yes` are treated as `True`. `0`, `f`, `false`, `off`, `n` and `no` are treated as `False` (they are case-insensitive). If the value is `None` (case-insensitive) both public and private flavors will be listed in a single request.

Response¶

Name	In	Type	Description
flavors	body	array	An array of flavor objects.
name	body	string	The display name of a flavor.
description	body	string	The description of the flavor.**New in version 2.55**
id	body	string	The ID of the flavor. While people often make this look like an int, this is really a string.
ram	body	integer	The amount of RAM a flavor has, in MiB.
disk	body	integer	The size of the root disk that will be created in GiB. If 0 the root disk will be set to exactly the size of the image used to deploy the instance. However, in this case filter scheduler cannot select the compute host based on the virtual image size. Therefore, 0 should only be used for volume booted instances or for testing purposes. Volume-backed instances can be enforced for flavors with zero root disk via the `os_compute_api:servers:create:zero_disk_flavor` policy rule.
vcpus	body	integer	The number of virtual CPUs that will be allocated to the server.
links	body	array	Links to the resources in question. See API Guide / Links and Referencesfor more info.
OS-FLV-EXT-DATA:ephemeral	body	integer	The size of the ephemeral disk that will be created, in GiB. Ephemeral disks may be written over on server state changes. So should only be used as a scratch space for applications that are aware of its limitations. Defaults to 0.
OS-FLV-DISABLED:disabled (Optional)	body	boolean	Whether or not the flavor has been administratively disabled. This is typically only visible to administrative users.
swap	body	integer	The size of a dedicated swap disk that will be allocated, in MiB. If 0 (the default), no dedicated swap disk will be created. Currently, the empty string (‘’) is used to represent 0.
rxtx_factor	body	float	The receive / transmit factor (as a float) that will be set on ports if the network backend supports the QOS extension. Otherwise it will be ignored. It defaults to 1.0.
os-flavor-access:is_public	body	boolean	Whether the flavor is public (available to all projects) or scoped to a set of projects. Default is True if not specified.
extra_specs (Optional)	body	object	A dictionary of the flavor’s extra-specs key-and-value pairs. This will only be included if the user is allowed by policy to index flavor extra_specs.**New in version 2.61**

Example List Flavors With Details (v2.61)


```
{
    "flavors": [
        {
            "OS-FLV-DISABLED:disabled": false,
            "disk": 1,
            "OS-FLV-EXT-DATA:ephemeral": 0,
            "os-flavor-access:is_public": true,
            "id": "1",
            "links": [
                {
                    "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/flavors/1",
                    "rel": "self"
                },
                {
                    "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/1",
```

```
                "rel": "bookmark"
            }
        ],
        "name": "m1.tiny",
        "ram": 512,
        "swap": "",
        "vcpus": 1,
        "rxtx_factor": 1.0,
        "description": null,
        "extra_specs": {}
    },
    {
        "OS-FLV-DISABLED:disabled": false,
        "disk": 20,
        "OS-FLV-EXT-DATA:ephemeral": 0,
        "os-flavor-access:is_public": true,
        "id": "2",
        "links": [
            {
                "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/flavors/2",
                "rel": "self"
            },
            {
                "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/2",
                "rel": "bookmark"
            }
        ],
        "name": "m1.small",
        "ram": 2048,
        "swap": "",
        "vcpus": 1,
        "rxtx_factor": 1.0,
        "description": null,
        "extra_specs": {}
    },
    {
        "OS-FLV-DISABLED:disabled": false,
        "disk": 40,
        "OS-FLV-EXT-DATA:ephemeral": 0,
        "os-flavor-access:is_public": true,
        "id": "3",
        "links": [
            {
                "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/flavors/3",
                "rel": "self"
            },
            {
                "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/3",
                "rel": "bookmark"
            }
        ],
        "name": "m1.medium",
        "ram": 4096,
        "swap": "",
        "vcpus": 2,
        "rxtx_factor": 1.0,
        "description": null,
        "extra_specs": {}
    },
    {
        "OS-FLV-DISABLED:disabled": false,
        "disk": 80,
        "OS-FLV-EXT-DATA:ephemeral": 0,
        "os-flavor-access:is_public": true,
        "id": "4",
        "links": [
            {
                "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/flavors/4",
                "rel": "self"
            },
            {
                "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/4",
                "rel": "bookmark"
            }
        ],
        "name": "m1.large",
        "ram": 8192,
        "swap": "",
        "vcpus": 4,
```

```
            "rxtx_factor": 1.0,
            "description": null,
            "extra_specs": {}
        },
        {
            "OS-FLV-DISABLED:disabled": false,
            "disk": 160,
            "OS-FLV-EXT-DATA:ephemeral": 0,
            "os-flavor-access:is_public": true,
            "id": "5",
            "links": [
                {
                    "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/flavors/5",
                    "rel": "self"
                },
                {
                    "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/5",
                    "rel": "bookmark"
                }
            ],
            "name": "m1.xlarge",
            "ram": 16384,
            "swap": "",
            "vcpus": 8,
            "rxtx_factor": 1.0,
            "description": null,
            "extra_specs": {}
        },
        {
            "OS-FLV-DISABLED:disabled": false,
            "disk": 1,
            "OS-FLV-EXT-DATA:ephemeral": 0,
            "os-flavor-access:is_public": true,
            "id": "6",
            "links": [
                {
                    "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/flavors/6",
                    "rel": "self"
                },
                {
                    "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/6",
                    "rel": "bookmark"
                }
            ],
            "name": "m1.tiny.specs",
            "ram": 512,
            "swap": "",
            "vcpus": 1,
            "rxtx_factor": 1.0,
            "description": null,
            "extra_specs": {
                "hw:mem_page_size": "2048",
                "hw:cpu_policy": "dedicated"
            }
        },
        {
            "OS-FLV-DISABLED:disabled": false,
            "disk": 20,
            "OS-FLV-EXT-DATA:ephemeral": 0,
            "os-flavor-access:is_public": true,
            "id": "7",
            "links": [
                {
                    "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/flavors/7",
                    "rel": "self"
                },
                {
                    "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/7",
                    "rel": "bookmark"
                }
            ],
            "name": "m1.small.description",
            "ram": 2048,
            "swap": "",
            "vcpus": 1,
            "rxtx_factor": 1.0,
            "description": "test description",
            "extra_specs": {
                "key1": "value1",
```

```
            "key2": "value2"
        }
    }
  ]
}
```

Show Flavor Details GET /flavors/{flavor_id}

Shows details for a flavor.

Normal response codes: 200

Error response codes: unauthorized(401), forbidden(403), itemNotFound(404)

Request¶

Name	In	Type	Description
flavor_id	path	string	The ID of the flavor.

Response¶

Name	In	Type	Description
flavor	body	object	The ID and links for the flavor for your server instance. A flavor is a combination of memory, disk size, and CPUs.
name	body	string	The display name of a flavor.
description	body	string	The description of the flavor.**New in version 2.55**
id	body	string	The ID of the flavor. While people often make this look like an int, this is really a string.
ram	body	integer	The amount of RAM a flavor has, in MiB.
disk	body	integer	The size of the root disk that will be created in GiB. If 0 the root disk will be set to exactly the size of the image used to deploy the instance. However, in this case filter scheduler cannot select the compute host based on the virtual image size. Therefore, 0 should only be used for volume booted instances or for testing purposes. Volume-backed instances can be enforced for flavors with zero root disk via the `os_compute_api:servers:create:zero_disk_flavor` policy rule.
vcpus	body	integer	The number of virtual CPUs that will be allocated to the server.
links	body	array	Links to the resources in question. See API Guide / Links and Referencesfor more info.
OS-FLV-EXT-DATA:ephemeral	body	integer	The size of the ephemeral disk that will be created, in GiB. Ephemeral disks may be written over on server state changes. So should only be used as a scratch space for applications that are aware of its limitations. Defaults to 0.
OS-FLV-DISABLED:disabled (Optional)	body	boolean	Whether or not the flavor has been administratively disabled. This is typically only visible to administrative users.
swap	body	integer	The size of a dedicated swap disk that will be allocated, in MiB. If 0 (the default), no dedicated swap disk will be created. Currently, the empty string ('') is used to represent 0.
rxtx_factor	body	float	The receive / transmit factor (as a float) that will be set on ports if the network backend supports the QOS extension. Otherwise it will be ignored. It defaults to 1.0.
os-flavor-access:is_public	body	boolean	Whether the flavor is public (available to all projects) or scoped to a set of projects. Default is True if not specified.
extra_specs (Optional)	body	object	A dictionary of the flavor's extra-specs key-and-value pairs. This will only be included if the user is allowed by policy to index flavor extra_specs.**New in version 2.61**

Example Show Flavor Details (v2.61)

```
{
    "flavor": {
        "OS-FLV-DISABLED:disabled": false,
        "disk": 20,
        "OS-FLV-EXT-DATA:ephemeral": 0,
        "os-flavor-access:is_public": true,
        "id": "7",
        "links": [
            {
                "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/flavors/7",
                "rel": "self"
            },
            {
                "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/7",
                "rel": "bookmark"
            }
        ],
        "name": "m1.small.description",
        "ram": 2048,
        "swap": "",
        "vcpus": 1,
        "rxtx_factor": 1.0,
        "description": "test description",
        "extra_specs": {
            "key1": "value1",
            "key2": "value2"
        }
    }
}
```

Delete Flavor DELETE /flavors/{flavor_id}

Deletes a flavor.

This is typically an admin only action. Deleting a flavor that is in use by existing servers is not recommended as it can cause incorrect data to be returned to the user under some operations.

Normal response codes: 202

Error response codes: unauthorized(401), forbidden(403), itemNotFound(404)

Request¶

Name	In	Type	Description
flavor_id	path	string	The ID of the flavor.

Response¶

No body content is returned on a successful DELETE.

Usage reports (os-simple-tenant-usage)¶

Reports usage statistics of compute and storage resources periodically for an individual tenant or all tenants. The usage statistics will include all instances' CPU, memory and local disk during a specific period.

Microversion 2.40 added pagination (and `next` links) to the usage statistics via optional `limit` and `marker` query parameters. If `limit` isn't provided, the configurable `max_limit` will be used which currently defaults to 1000. Older microversions will not accept these new paging query parameters, but they will start to silently limit by `max_limit`.

```
/os-simple-tenant-usage?limit={limit}&marker={instance_uuid}
/os-simple-tenant-usage/{tenant_id}?limit={limit}&marker={instance_uuid}
```

> Note
>
> A tenant's usage statistics may span multiple pages when the number of instances exceeds `limit`, and API consumers will need to stitch together the aggregate results if they still want totals for all instances in a specific time window, grouped by tenant.

List Tenant Usage Statistics For All Tenants GET /os-simple-tenant-usage

Lists usage statistics for all tenants.

Normal response codes: 200

Error response codes: badRequest(400), unauthorized(401), forbidden(403)

Request¶

Name	In	Type	Description
detailed (Optional)	query	integer	Specify the `detailed=1` query parameter to get detail information ('server_usages' information).
end (Optional)	query	string	The ending time to calculate usage statistics on compute and storage resources. The date and time stamp format is any of the following ones: `CCYY-MM-DDThh:mm:ss` For example, `2015-08-27T09:49:58`. `CCYY-MM-DDThh:mm:ss.NNNNNN` For example, `2015-08-27T09:49:58.123456`. `CCYY-MM-DD hh:mm:ss.NNNNNN` For example, `2015-08-27 09:49:58.123456`. If you omit this parameter, the current time is used.
start (Optional)	query	string	The beginning time to calculate usage statistics on compute and storage resources. The date and time stamp format is any of the following ones: `CCYY-MM-DDThh:mm:ss` For example, `2015-08-27T09:49:58`. `CCYY-MM-DDThh:mm:ss.NNNNNN` For example, `2015-08-27T09:49:58.123456`. `CCYY-MM-DD hh:mm:ss.NNNNNN` For example, `2015-08-27 09:49:58.123456`. If you omit this parameter, the current time is used.
limit (Optional)	query	integer	Requests a page size of items. Calculate usage for the limited number of instances. Use the `limit` parameter to make an initial limited request and use the last-seen instance UUID from the response as the `marker` parameter value in a subsequent limited request.**New in version 2.40**
marker (Optional)	query	string	The last-seen item. Use the `limit` parameter to make an initial limited request and use the last-seen instance UUID from the response as the `marker` parameter value in a subsequent limited request.**New in version 2.40**

Response¶

Name	In	Type	Description
tenant_usages	body	array	A list of the tenant usage objects.
start	body	string	The beginning time to calculate usage statistics on compute and storage resources. The date and time stamp format is as follows: `CCYY-MM-DDThh:mm:ss.NNNNNN` For example, `2015-08-27T09:49:58.123456`.
stop	body	string	The ending time to calculate usage statistics on compute and storage resources. The date and time stamp format is as follows: `CCYY-MM-DDThh:mm:ss.NNNNNN` For example, `2015-08-27T09:49:58.123456`.
tenant_id	body	string	The UUID of the tenant in a multi-tenancy cloud.
total_hours	body	float	The total duration that servers exist (in hours).
total_local_gb_usage	body	float	Multiplying the server disk size (in GiB) by hours the server exists, and then adding that all together for each server.
total_memory_mb_usage	body	float	Multiplying the server memory size (in MiB) by hours the server exists, and then adding that all together for each server.
total_vcpus_usage	body	float	Multiplying the number of virtual CPUs of the server by hours the server exists, and then adding that all together for each server.
server_usages (Optional)	body	array	A list of the server usage objects.
server_usages.ended_at (Optional)	body	string	The date and time when the server was deleted.The date and time stamp format is as follows: `CCYY-MM-DDThh:mm:ss.NNNNNN` For example, `2015-08-27T09:49:58.123456`. If the server hasn't been deleted yet, its value is `null`.
server_usages.flavor (Optional)	body	string	The display name of a flavor.
server_usages.hours (Optional)	body	float	The duration that the server exists (in hours).
server_usages.instance_id (Optional)	body	string	The UUID of the server.
server_usages.local_gb (Optional)	body	integer	The sum of the root disk size of the server and the ephemeral disk size of it (in GiB).
server_usages.memory_mb (Optional)	body	integer	The memory size of the server (in MiB).
server_usages.name (Optional)	body	string	The server name.
server_usages.started_at (Optional)	body	string	The date and time when the server was launched.The date and time stamp format is as follows: `CCYY-MM-DDThh:mm:ss.NNNNNN` For example, `2015-08-27T09:49:58.123456`.
server_usages.state (Optional)	body	string	The VM state.
server_usages.tenant_id (Optional)	body	string	The UUID of the tenant in a multi-tenancy cloud.
server_usages.uptime (Optional)	body	integer	The uptime of the server.
server_usages.vcpus (Optional)	body	integer	The number of virtual CPUs that the server uses.
tenant_usages_links (Optional)	body	array	Links pertaining to usage. See API Guide / Links and References for more info.**New in version 2.40**

Example List Tenant Usage For All Tenants (v2.40): JSON response

If the `detailed` query parameter is not specified or is set to other than 1 (e.g. `detailed=0`), the response is as follows:

```
{
    "tenant_usages": [
        {
            "start": "2012-10-08T21:10:44.587336",
            "stop": "2012-10-08T22:10:44.587336",
            "tenant_id": "6f70656e737461636b20342065766572",
            "total_hours": 1.0,
            "total_local_gb_usage": 1.0,
            "total_memory_mb_usage": 512.0,
            "total_vcpus_usage": 1.0
        }
    ],
    "tenant_usages_links": [
        {
            "href": "http://openstack.example.com/v2.1/6f70656e737461636b20342065766572/os-simple-tenant-
usage?end=2016-10-12+18%3A22%3A04.868106&limit=1&marker=1f1deceb-17b5-4c04-84c7-e0d4499c8fe0&start=2016-10-
12+18%3A22%3A04.868106",
            "rel": "next"
        }
    ]
}
```

If the `detailed` query parameter is set to one (`detailed=1`), the response includes `server_usages` information for each tenant. The response is as follows:

```
{
    "tenant_usages": [
        {
            "start": "2012-10-08T20:10:44.587336",
            "stop": "2012-10-08T21:10:44.587336",
            "tenant_id": "6f70656e737461636b20342065766572",
            "total_hours": 1.0,
            "total_local_gb_usage": 1.0,
            "total_memory_mb_usage": 512.0,
            "total_vcpus_usage": 1.0,
            "server_usages": [
                {
                    "ended_at": null,
                    "flavor": "m1.tiny",
                    "hours": 1.0,
                    "instance_id": "1f1deceb-17b5-4c04-84c7-e0d4499c8fe0",
                    "local_gb": 1,
                    "memory_mb": 512,
                    "name": "instance-2",
                    "started_at": "2012-10-08T20:10:44.541277",
                    "state": "active",
                    "tenant_id": "6f70656e737461636b20342065766572",
                    "uptime": 3600,
                    "vcpus": 1
                }
            ]
        }
    ],
    "tenant_usages_links": [
        {
            "href": "http://openstack.example.com/v2.1/6f70656e737461636b20342065766572/os-simple-tenant-
usage?detailed=1&end=2016-10-12+18%3A22%3A04.868106&limit=1&marker=1f1deceb-17b5-4c04-84c7-
e0d4499c8fe0&start=2016-10-12+18%3A22%3A04.868106",
            "rel": "next"
        }
    ]
}
```

Show Usage Statistics For Tenant GET /os-simple-tenant-usage/{tenant_id}

Shows usage statistics for a tenant.

Normal response codes: 200

Error response codes: badRequest(400), unauthorized(401), forbidden(403)

Request¶

Name	In	Type	Description
tenant_id	path	string	The UUID of the tenant in a multi-tenancy cloud.
end (Optional)	query	string	The ending time to calculate usage statistics on compute and storage resources. The date and time stamp format is any of the following ones: `CCYY-MM-DDThh:mm:ss` For example, `2015-08-27T09:49:58`. `CCYY-MM-DDThh:mm:ss.NNNNNN` For example, `2015-08-27T09:49:58.123456`. `CCYY-MM-DD hh:mm:ss.NNNNNN` For example, `2015-08-27 09:49:58.123456`. If you omit this parameter, the current time is used.
start (Optional)	query	string	The beginning time to calculate usage statistics on compute and storage resources. The date and time stamp format is any of the following ones: `CCYY-MM-DDThh:mm:ss` For example, `2015-08-27T09:49:58`. `CCYY-MM-DDThh:mm:ss.NNNNNN` For example, `2015-08-27T09:49:58.123456`. `CCYY-MM-DD hh:mm:ss.NNNNNN` For example, `2015-08-27 09:49:58.123456`. If you omit this parameter, the current time is used.
limit (Optional)	query	integer	Requests a page size of items. Calculate usage for the limited number of instances. Use the `limit` parameter to make an initial limited request and use the last-seen instance UUID from the response as the `marker` parameter value in a subsequent limited request.**New in version 2.40**
marker (Optional)	query	string	The last-seen item. Use the `limit` parameter to make an initial limited request and use the last-seen instance UUID from the response as the `marker` parameter value in a subsequent limited request.**New in version 2.40**

Response¶

Name	In	Type	Description
tenant_usage	body	object	The tenant usage object.
server_usages	body	array	A list of the server usage objects.
server_usages.ended_at	body	string	The date and time when the server was deleted.The date and time stamp format is as follows: `CCYY-MM-DDThh:mm:ss.NNNNNN` For example, `2015-08-27T09:49:58.123456`. If the server hasn't been deleted yet, its value is `null`.
server_usages.flavor	body	string	The display name of a flavor.
server_usages.hours	body	float	The duration that the server exists (in hours).
server_usages.instance_id	body	string	The UUID of the server.
server_usages.local_gb	body	integer	The sum of the root disk size of the server and the ephemeral disk size of it (in GiB).
server_usages.memory_mb	body	integer	The memory size of the server (in MiB).
server_usages.name	body	string	The server name.
server_usages.started_at	body	string	The date and time when the server was launched.The date and time stamp format is as follows: `CCYY-MM-DDThh:mm:ss.NNNNNN` For example, `2015-08-27T09:49:58.123456`.
server_usages.state	body	string	The VM state.
server_usages.tenant_id	body	string	The UUID of the tenant in a multi-tenancy cloud.
server_usages.uptime	body	integer	The uptime of the server.
server_usages.vcpus	body	integer	The number of virtual CPUs that the server uses.
start	body	string	The beginning time to calculate usage statistics on compute and storage resources. The date and time stamp format is as follows: `CCYY-MM-DDThh:mm:ss.NNNNNN` For example, `2015-08-27T09:49:58.123456`.
stop	body	string	The ending time to calculate usage statistics on compute and storage resources. The date and time stamp format is as follows: `CCYY-MM-DDThh:mm:ss.NNNNNN` For example, `2015-08-27T09:49:58.123456`.
tenant_id	body	string	The UUID of the tenant in a multi-tenancy cloud.
total_hours	body	float	The total duration that servers exist (in hours).
total_local_gb_usage	body	float	Multiplying the server disk size (in GiB) by hours the server exists, and then adding that all together for each server.
total_memory_mb_usage	body	float	Multiplying the server memory size (in MiB) by hours the server exists, and then adding that all together for each server.
total_vcpus_usage	body	float	Multiplying the number of virtual CPUs of the server by hours the server exists, and then adding that all together for each server.
tenant_usage_links (Optional)	body	array	Links pertaining to usage. See API Guide / Links and References for more info.**New in version 2.40**

Example Show Usage Details For Tenant (v2.40): JSON response

```
{
    "tenant_usage": {
        "server_usages": [
            {
                "ended_at": null,
                "flavor": "m1.tiny",
                "hours": 1.0,
                "instance_id": "1f1deceb-17b5-4c04-84c7-e0d4499c8fe0",
                "local_gb": 1,
                "memory_mb": 512,
                "name": "instance-2",
                "started_at": "2012-10-08T20:10:44.541277",
                "state": "active",
                "tenant_id": "6f70656e737461636b20342065766572",
                "uptime": 3600,
                "vcpus": 1
            }
        ],
        "start": "2012-10-08T20:10:44.587336",
        "stop": "2012-10-08T21:10:44.587336",
        "tenant_id": "6f70656e737461636b20342065766572",
        "total_hours": 1.0,
        "total_local_gb_usage": 1.0,
        "total_memory_mb_usage": 512.0,
        "total_vcpus_usage": 1.0
    },
    "tenant_usage_links": [
        {
            "href": "http://openstack.example.com/v2.1/6f70656e737461636b20342065766572/os-simple-tenant-
usage/6f70656e737461636b20342065766572?end=2016-10-12+18%3A22%3A04.868106&limit=1&marker=1f1deceb-17b5-
4c04-84c7-e0d4499c8fe0&start=2016-10-12+18%3A22%3A04.868106",
            "rel": "next"
```

```
        }
    ]
}
```

Port interfaces (servers, os-interface)¶

List port interfaces, show port interface details of the given server. Create a port interface and uses it to attach a port to the given server, detach a port interface from the given server.

List Port Interfaces GET /servers/{server_id}/os-interface

Lists port interfaces that are attached to a server.

Normal response codes: 200

Error response codes: unauthorized(401), forbidden(403), itemNotFound(404), NotImplemented(501)

Request¶

Name	In	Type	Description
server_id	path	string	The UUID of the server.

Response¶

Name	In	Type	Description
interfaceAttachments	body	array	List of the interface attachments.
port_state	body	string	The port state.
fixed_ips	body	array	Fixed IP addresses with subnet IDs.
ip_address	body	string	The IP address.
subnet_id	body	string	The UUID of the subnet.
mac_addr	body	string	The MAC address.
net_id	body	string	The network ID.
port_id	body	string	The port ID.

Example List Port Interfaces: JSON response

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

Create Interface POST /servers/{server_id}/os-interface

Creates a port interface and uses it to attach a port to a server.

Normal response codes: 200

Error response codes: badRequest(400), unauthorized(401), forbidden(403), itemNotFound(404), conflict(409), computeFault(500), NotImplemented(501)

Request¶

Name	In	Type	Description
server_id	path	string	The UUID of the server.
interfaceAttachment	body	string	Specify the `interfaceAttachment` action in the request body.
port_id (Optional)	body	string	The ID of the port for which you want to create an interface. The `net_id` and `port_id` parameters are mutually exclusive. If you do not specify the `port_id` parameter, the OpenStack Networking API v2.0 allocates a port and creates an interface for it on the network.
net_id (Optional)	body	string	The ID of the network for which you want to create a port interface. The `net_id` and `port_id` parameters are mutually exclusive. If you do not specify the `net_id` parameter, the OpenStack Networking API v2.0 uses the network information cache that is associated with the instance.
fixed_ips (Optional)	body	array	Fixed IP addresses. If you request a specific fixed IP address without a `net_id`, the request returns a `Bad Request(400)` response code.
ip_address	body	string	The IP address. It is required when `fixed_ips` is specified.
tag (Optional)	body	string	A device role tag that can be applied to a network interface when attaching it to the VM. The guest OS of a server that has devices tagged in this manner can access hardware metadata about the tagged devices from the metadata API and on the config drive, if enabled.**New in version 2.49**

Example Create Interface: JSON request

Create interface with `net_id` and `fixed_ips`.

```
{
    "interfaceAttachment": {
        "fixed_ips": [
            {
                "ip_address": "192.168.1.3"
            }
        ],
        "net_id": "3cb9bc59-5699-4588-a4b1-b87f96708bc6"
    }
}
```

Create interface with `port_id`.

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

Example Create Tagged Interface (v2.49): JSON request

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

Response¶

Name	In	Type	Description
interfaceAttachment	body	object	The interface attachment.
fixed_ips	body	array	Fixed IP addresses with subnet IDs.
ip_address	body	string	The IP address.
subnet_id	body	string	The UUID of the subnet.
mac_addr	body	string	The MAC address.
net_id	body	string	The network ID.
port_id	body	string	The port ID.
port_state	body	string	The port state.

Example Create Interface: JSON response

```
{
```

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


Show Port Interface Details GET /servers/{server_id}/os-interface/{port_id}

Shows details for a port interface that is attached to a server.

Normal response codes: 200

Error response codes: unauthorized(401), forbidden(403), itemNotFound(404)

Request¶

Name	In	Type	Description
server_id	path	string	The UUID of the server.
port_id	path	string	The UUID of the port.

Response¶

Name	In	Type	Description
interfaceAttachment	body	object	The interface attachment.
port_state	body	string	The port state.
fixed_ips	body	array	Fixed IP addresses with subnet IDs.
ip_address	body	string	The IP address.
subnet_id	body	string	The UUID of the subnet.
mac_addr	body	string	The MAC address.
net_id	body	string	The network ID.
port_id	body	string	The port ID.

Example Show Port Interface Details: JSON response


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


Detach Interface DELETE /servers/{server_id}/os-interface/{port_id}

Detaches a port interface from a server.

Normal response codes: 202

Error response codes: unauthorized(401), forbidden(403), itemNotFound(404), conflict(409), NotImplemented(501)

Request¶

Name	In	Type	Description
server_id	path	string	The UUID of the server.
port_id	path	string	The UUID of the port.