

OpenStack

Compute API

(August 6, 2016)



docs.openstack.org

Compute API

Table of Contents

- [Compute API](#)
- [API Versions](#)
 - [List All Major Versions](#)
 - [Show Details of Specific API Version](#)
- [Service URLs](#)
- [Servers \(servers\)](#)
 - [List Servers](#)
 - [Create Server](#)
 - [Create Multiple Servers](#)
 - [List Servers Detailed](#)
 - [Show Server Details](#)
 - [Update Server](#)
 - [Delete Server](#)
- [Servers - run an action \(servers, action\)](#)
 - [Add \(Associate\) Floating Ip \(addFloatingIp Action\)](#)
 - [Add Security Group To A Server \(addSecurityGroup Action\)](#)
 - [Change Administrative Password \(changePassword Action\)](#)
 - [Confirm Resized Server \(confirmResize Action\)](#)
 - [Create Image \(createImage Action\)](#)
 - [Lock Server \(lock Action\)](#)
 - [Pause Server \(pause Action\)](#)
 - [Reboot Server \(reboot Action\)](#)
 - [Rebuild Server \(rebuild Action\)](#)
 - [Remove \(Disassociate\) Floating Ip \(removeFloatingIp Action\)](#)
 - [Remove Security Group From A Server \(removeSecurityGroup Action\)](#)
 - [Rescue Server \(rescue Action\)](#)
 - [Resize Server \(resize Action\)](#)
 - [Resume Suspended Server \(resume Action\)](#)
 - [Revert Resized Server \(revertResize Action\)](#)
 - [Start Server \(os-start Action\)](#)
 - [Stop Server \(os-stop Action\)](#)
 - [Suspend Server \(suspend Action\)](#)
 - [Unlock Server \(unlock Action\)](#)
 - [Unpause Server \(unpause Action\)](#)
 - [Unrescue Server \(unrescue Action\)](#)
 - [Add \(Associate\) Fixed Ip \(Addfixedip Action\)](#)
 - [Remove \(Disassociate\) Fixed Ip \(Removefixedip Action\)](#)
 - [Evacuate Server \(evacuate Action\)](#)
 - [Force-Delete Server \(forceDelete Action\)](#)
 - [Restore Soft-Deleted Instance \(restore Action\)](#)
 - [Show Console Output \(Os-Getconsoleoutput Action\)](#)
 - [Get Rdp Console \(os-getRDPCConsole Action\)](#)
 - [Get Serial Console \(os-getSerialConsole Action\)](#)
 - [Get Spice Console \(os-getSPICEConsole Action\)](#)
 - [Get Vnc Console \(os-getVNCCConsole Action\)](#)
 - [Shelve Server \(shelve Action\)](#)

- [Shelf-Offload \(Remove\) Server \(shelveOffload Action\)](#)
 - [Unshelve \(Restore\) Shelved Server \(unshelve Action\)](#)
 - [Trigger Crash Dump In Server](#)
- [Servers - run an administrative action \(servers, action\)](#)
 - [Create Server Back Up \(createBackup Action\)](#)
 - [Inject Network Information \(injectNetworkInfo Action\)](#)
 - [Migrate Server \(migrate Action\)](#)
 - [Live-Migrate Server \(os-migrateLive Action\)](#)
 - [Reset Networking On A Server \(resetNetwork Action\)](#)
 - [Reset Server State \(os-resetState Action\)](#)
- [Servers Security Groups \(servers, os-security-groups\)](#)
 - [List Security Groups By Server](#)
- [Servers diagnostics \(servers, diagnostics\)](#)
 - [Show Server Diagnostics](#)
- [Servers IPs \(servers, ips\)](#)
 - [List Ips](#)
 - [Show Ip Details](#)
- [Server metadata \(servers, metadata\)](#)
 - [List All Metadata](#)
 - [Update Metadata Items](#)
 - [Create Or Replace Metadata Items](#)
 - [Show Metadata Item Details](#)
 - [Create Or Update Metadata Item](#)
 - [Delete Metadata Item](#)
- [Servers actions \(servers, os-instance-actions\)](#)
 - [List Actions For Server](#)
 - [Show Server Action Details](#)
- [Port interfaces \(servers, os-interface\)](#)
 - [List Port Interfaces](#)
 - [Create Interface](#)
 - [Show Port Interface Details](#)
 - [Detach Interface](#)
- [Servers password \(servers, os-server-password\)](#)
 - [Show Server Password](#)
 - [Clear Admin Password](#)
- [Servers virtual interfaces \(servers, os-virtual-interfaces\)](#)
 - [List Virtual Interfaces](#)
- [Servers with volume attachments \(servers, os-volume_attachments\)](#)
 - [List volume attachments for an instance](#)
 - [Attach a volume to an instance](#)
 - [Show a detail of a volume attachment](#)
 - [Update a volume attachment](#)
 - [Detach a volume from an instance](#)
- [Flavors](#)
 - [List Flavors](#)
 - [Create Flavor](#)
 - [List Flavors With Details](#)
 - [Show Flavor Details](#)
 - [Delete Flavor](#)
- [Flavors access \(flavors, os-flavor-access\)](#)
 - [List Flavor Access Information For Given Flavor](#)
 - [Add Flavor Access To Tenant \(addTenantAccess Action\)](#)

- [Remove Flavor Access From Tenant \(removeTenantAccess Action\)](#)
- [Flavors extra-specs \(flavors, os-flavor-extra-specs\)](#)
 - [List Extra Specs For A Flavor](#)
 - [Create Extra Specs For A Flavor](#)
 - [Show An Extra Spec For A Flavor](#)
 - [Update An Extra Spec For A Flavor](#)
 - [Delete An Extra Spec For A Flavor](#)
- [Keypairs \(keypairs\)](#)
 - [List Keypairs](#)
 - [Create Or Import Keypair](#)
 - [Show Keypair Details](#)
 - [Delete Keypair](#)
- [Limits \(limits\)](#)
 - [Show Rate And Absolute Limits](#)
- [Guest agents \(os-agents\)](#)
 - [List Agent Builds](#)
 - [Create Agent Build](#)
 - [Update Agent Build](#)
 - [Delete Agent Build](#)
- [Host aggregates \(os-aggregates\)](#)
 - [List Aggregates](#)
 - [Create Aggregate](#)
 - [Show Aggregate Details](#)
 - [Update Aggregate](#)
 - [Delete Aggregate](#)
 - [Add Host](#)
 - [Remove Host](#)
 - [Create Or Update Aggregate Metadata](#)
- [Assisted volume snapshots \(os-assisted-volume-snapshots\)](#)
 - [Create Assisted Volume Snapshots](#)
 - [Delete Assisted Volume Snapshot](#)
- [Availability zones \(os-availability-zone\)](#)
 - [Get Availability Zone Information](#)
 - [Get Detailed Availability Zone Information](#)
- [Cells \(os-cells, capacities\)](#)
 - [List Cells](#)
 - [Create Cell](#)
 - [Capacities](#)
 - [List Cells With Details](#)
 - [Info For This Cell](#)
 - [Show Cell Data](#)
 - [Update a Cell](#)
 - [Delete a Cell](#)
 - [Show Cell Capacities](#)
- [Cloudpipe \(os-cloudpipe\)](#)
 - [List Cloudpipes](#)
 - [Create Cloudpipe](#)
 - [Update Cloudpipe](#)
- [Server consoles \(servers, os-consoles, os-console-auth-token\)](#)
 - [Lists Consoles](#)
 - [Create Console](#)
 - [Show Console Details](#)

- [Delete Console](#)
 - [Show Console Authentication Token](#)
- [Hosts \(os-hosts\)](#)
 - [List Hosts](#)
 - [Show Host Details](#)
 - [Update Host status](#)
 - [Reboot Host](#)
 - [Shut Down Host](#)
 - [Start Host](#)
- [Hypervisors \(os-hypervisors\)](#)
 - [List Hypervisors](#)
 - [List Hypervisors Details](#)
 - [Show Hypervisor Statistics](#)
 - [Show Hypervisor Details](#)
 - [Show Hypervisor Uptime](#)
 - [Search Hypervisor](#)
 - [List Hypervisor Servers](#)
- [Server usage audit log \(os-instance-usage-audit-log\)](#)
 - [List Server Usage Audits For An Tenant](#)
 - [List Usage Audits Before Specified Time](#)
- [Migrations \(os-migrations\) \(frozen\)](#)
 - [List Migrations](#)
- [Quota sets \(os-quota-sets\)](#)
 - [Show A Quota](#)
 - [Update Quotas](#)
 - [Revert Quotas To Defaults](#)
 - [List Default Quotas For Tenant](#)
 - [Show The Detail of Quota](#)
- [Server groups \(os-server-groups\)](#)
 - [List Server Groups](#)
 - [Create Server Group](#)
 - [Show Server Group Details](#)
 - [Delete Server Group](#)
- [Server tags \(servers, tags\)](#)
 - [List Tags](#)
 - [Replace Tags](#)
 - [Delete All Tags](#)
 - [Check Tag Existence](#)
 - [Add a Single Tag](#)
 - [Delete a Single Tag](#)
- [Compute services \(os-services\)](#)
 - [List Compute Services](#)
 - [Disable Scheduling For A Compute Service](#)
 - [Log Disabled Compute Service Information](#)
 - [Enable Scheduling For A Compute Service](#)
 - [Update Forced Down](#)
 - [Delete Compute Service](#)
- [Usage reports \(os-simple-tenant-usage\)](#)
 - [List Tenant Usage For All Tenants](#)
 - [Show Usage Details For Tenant](#)
- [Create external events \(os-server-external-events\)](#)
 - [Run Events](#)

- [Extensions \(extensions\) \(DEPRECATED\)](#)
 - [List Extensions](#)
 - [Show Extension Details](#)
- [Root certificates \(os-certificates\) \(DEPRECATED\)](#)
 - [Create Root Certificate](#)
 - [Show Root Certificate Details](#)
- [Networks \(os-networks\) \(DEPRECATED\)](#)
 - [List Networks](#)
 - [Create Network](#)
 - [Add Network](#)
 - [Show Network Details](#)
 - [Delete Network](#)
 - [Associate Host \(DEPRECATED\)](#)
 - [Disassociate Network \(DEPRECATED\)](#)
 - [Disassociate Host \(DEPRECATED\)](#)
 - [Disassociate Project \(DEPRECATED\)](#)
- [Volume extension \(os-volumes, os-snapshots\) \(DEPRECATED\)](#)
 - [List Volumes](#)
 - [Create Volume](#)
 - [List Volumes With Details](#)
 - [Show Volume Details](#)
 - [Delete Volume](#)
 - [List Snapshots](#)
 - [Create Snapshot](#)
 - [List Snapshots With Details](#)
 - [Show Snapshot Details](#)
 - [Delete Snapshot](#)
- [Images \(DEPRECATED\)](#)
 - [List Images](#)
 - [List Images With Details](#)
 - [Show Image Details](#)
 - [Delete Image](#)
 - [List Image Metadata](#)
 - [Create Image Metadata](#)
 - [Update Image Metadata](#)
 - [Show Image Metadata Item](#)
 - [Create Or Update Image Metadata Item](#)
 - [Delete Image Metadata Item](#)
- [Bare metal nodes \(os-baremetal-nodes\) \(DEPRECATED\)](#)
 - [List Bare Metal Nodes](#)
 - [Show Bare Metal Node Details](#)
- [Project networks \(os-tenant-networks\) \(DEPRECATED\)](#)
 - [List Project Networks](#)
 - [Create Project Network](#)
 - [Show Project Network Details](#)
 - [Delete Project Network](#)
- [Fixed IPs \(os-fixed-ips\) \(DEPRECATED\)](#)
 - [Show Fixed Ip Details](#)
 - [Reserve Or Release A Fixed Ip](#)
- [Floating IP DNS records \(os-floating-ip-dns\) \(DEPRECATED\)](#)
 - [List Dns Domains](#)
 - [Create Or Update Dns Domain](#)

- [Delete Dns Domain](#)
- [List Dns Entries](#)
- [Find Unique Dns Entry](#)
- [Create Or Update Dns Entry](#)
- [Delete Dns Entry](#)
- [Floating IP pools \(os-floating-ip-pools\) \(DEPRECATED\)](#)
 - [List Floating Ip Pools](#)
- [Floating IPs \(os-floating-ips\) \(DEPRECATED\)](#)
 - [List Floating Ip Addresses](#)
 - [Create \(Allocate\) Floating Ip Address](#)
 - [Show Floating Ip Address Details](#)
 - [Delete \(Deallocate\) Floating Ip Address](#)
- [Floating IPs bulk \(os-floating-ips-bulk\) \(DEPRECATED\)](#)
 - [List Floating Ips](#)
 - [Create Floating Ips](#)
 - [Bulk-Delete Floating Ips](#)
 - [List Floating Ips By Host](#)
- [Ping instances \(os-fping\) \(DEPRECATED\)](#)
 - [Ping Instances](#)
 - [Ping An Instance](#)
- [Security groups \(os-security-groups\) \(DEPRECATED\)](#)
 - [List Security Groups](#)
 - [Create Security Group](#)
 - [Show Security Group Details](#)
 - [Update Security Group](#)
 - [Delete Security Group](#)
- [Rules for default security group \(os-security-group-default-rules\) \(DEPRECATED\)](#)
 - [List Default Security Group Rules](#)
 - [Show Default Security Group Rule Details](#)
 - [Create Default Security Group Rule](#)
 - [Delete Default Security Group Rule](#)
- [Rules for security group \(os-security-group-rules\) \(DEPRECATED\)](#)
 - [Create Security Group Rule](#)
 - [Delete Security Group Rule](#)

API Versions

In order to bring new features to users over time, the Nova API supports versioning. There are two kinds of versions in Nova.

- ‘‘major versions’’, which have dedicated urls
- ‘‘microversions’’, which can be requested through the use of the `X-OpenStack-Nova-API-Version` header or since microversion 2.27 the `OpenStack-API-Version` header may also be used.

For more detail about Microversion, please reference: [Microversions](#)

The Version APIs work differently from other APIs as they *do not* require authentication.

GET

/

List All Major Versions

This fetches all the information about all known major API versions in the deployment. Links to more specific information will be provided for each API version, as well as information about supported min and max microversions.

Normal Response Codes: 200

Response

| Name | In | Type | Description |
|-------------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| versions | body | array | A list of version objects that describe the API versions available. |
| id | body | string | A common name for the version in question. Informative only, it has no real semantic meaning. |
| status | body | string | The status of this API version. This can be one of: <ul style="list-style-type: none">• CURRENT: this is the preferred version of the API to use• SUPPORTED: this is an older, but still supported version of the API• DEPRECATED: a deprecated version of the API that is slated for removal |
| links | body | array | Links to the resources in question. See API Guide / Links and References for more info. |
| version | body | string | If this version of the API supports microversions, the maximum microversion that is supported. This will be the empty string if microversions are not supported. |
| min_version | body | string | If this version of the API supports microversions, the minimum microversion that is supported. This will be the empty string if microversions are not supported. |

Note

The `updated` parameter in the response is vestigial and provides no useful information.

Response Example

This demonstrates the expected response from a bleeding edge server that supports up to the current microversion. When querying OpenStack environments you will typically find the current microversion on the v2.1 API is lower than listed below.

```
{
  "versions": [
    {
      "id": "v2.0",
      "links": [
        {
          "href": "http://openstack.example.com/v2/",
          "rel": "self"
        }
      ],
      "status": "SUPPORTED",
      "version": "",
      "min_version": "",
      "updated": "2011-01-21T11:33:21Z"
    },
    {
      "id": "v2.1",
      "links": [
        {
          "href": "http://openstack.example.com/v2.1/",
          "rel": "self"
        }
      ],
      "status": "CURRENT",
      "version": "2.36",
      "min_version": "2.1",
      "updated": "2013-07-23T11:33:21Z"
    }
  ]
}
```

GET

`/api_version`

Show Details of Specific API Version

This gets the details of a specific API at it's root. Nearly all this information exists at the API root, so this is mostly a redundant operation.

Normal Response Codes: 200

Request

| Name | In | Type | Description |
|-------------|------|--------|---------------------------------------------------------------|
| api_version | path | string | The API version as returned in the links from the GET / call. |

Response

| Name | In | Type | Description |
|-------------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| version | body | string | The version. |
| id | body | string | A common name for the version in question. Informative only, it has no real semantic meaning. |
| status | body | string | The status of this API version. This can be one of: <ul style="list-style-type: none">• CURRENT: this is the preferred version of the API to use• SUPPORTED: this is an older, but still supported version of the API• DEPRECATED: a deprecated version of the API that is slated for removal |
| links | body | array | Links to the resources in question. See API Guide / Links and References for more info. |
| version | body | string | If this version of the API supports microversions, the maximum microversion that is supported. This will be the empty string if microversions are not supported. |
| min_version | body | string | If this version of the API supports microversions, the minimum microversion that is supported. This will be the empty string if microversions are not supported. |

Note

The `updated` and `media-types` parameters in the response are vestigial and provide no useful information. They will probably be deprecated and removed in the future.

Response Example

This is an example of a `GET /v2.1` on a relatively current server.

```
{
  "version": {
    "id": "v2.1",
    "links": [
      {
        "href": "http://openstack.example.com/v2.1/",
        "rel": "self"
      },
      {
        "href": "http://docs.openstack.org/",
        "rel": "describedby",
        "type": "text/html"
      }
    ],
    "media-types": [
      {
        "base": "application/json",
        "type": "application/vnd.openstack.compute+json;version=2.1"
      }
    ],
    "status": "CURRENT",
    "version": "2.36",
    "min_version": "2.1",
    "updated": "2013-07-23T11:33:21Z"
  }
}
```

Service URLs

All API calls through the rest of this document require authentication with the OpenStack Identity service. They also required a base `service url` that is extracted from the Identity token of type `compute`. This will be the root url that every call below will be added to build a full path.

For instance, if the `service url` is `http://mycompute.pvt/compute/v2.1` then the full API call for `/servers` is `http://mycompute.pvt/compute/v2.1/servers`.

Depending on the deployment the compute service url might be http or https, a custom port, a custom path, and include your tenant id. The only way to know the urls for your deployment is by using the service catalog. The compute URL should never be hard coded in applications, even if they are only expected to work at a single site. It should always be discovered from the Identity token.

As such, for the rest of this document we will be using short hand where `GET /servers` really means `GET {your_compute_service_url}/servers`.

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

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.

GET

/servers

List 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.
- **BUILDING.** 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.
- **RESCUED.** The server is in rescue mode. A rescue image is running with the original server image attached.
- **RESIZED.** 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.
- **SOFT_DELETED.** The server is marked as deleted but the disk images are still available to restore.
- **STOPPED.** The server is powered off and the disk image still persists.
- **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.

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 <code>limit</code> parameter to make an initial limited request and use the ID of the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request. |
| marker (Optional) | query | string | The ID of the last-seen item. Use the <code>limit</code> parameter to make an initial limited request and use the ID of the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request. |
| sort_key (Optional) | query | string | Sorts by a server attribute. Default attribute is <code>created</code> . 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 server <code>sort_key</code> attribute. |
| sort_dir (Optional) | query | string | Sort direction. A valid value is <code>asc</code> (ascending) or <code>desc</code> (descending). Default is <code>desc</code> . 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 server <code>sort_key</code> attribute. |

| Name | In | Type | Description |
|---------------------------|-------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| changes-since (Optional) | query | string | <p>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.</p> <p>The date and time stamp format is ISO 8601:</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>The ±hh:mm value, if included, returns the time zone as an offset from UTC. For example, 2015-08-27T09:49:58-05:00. If you omit the time zone, the UTC time zone is assumed.</p> |
| image (Optional) | query | string | Filters the response by an image, as a UUID. |
| flavor (Optional) | query | string | Filters the response by a flavor, as a UUID. A flavor is a combination of memory, disk size, and CPUs. |
| 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 only bob, you can use a regular expression that matches the syntax of the underlying database server that is implemented for Compute, such as MySQL or PostgreSQL. |
| status (Optional) | query | string | Filters the response by a server status, as a string. For example, ACTIVE. |
| ip (Optional) | query | string | An IPv4 address to filter results by. |
| reservation_id (Optional) | query | string | A reservation id as returned by a servers multiple create call. |
| all_tenants (Optional) | query | integer | Specify the all_tenants=1 query parameter to list all instances for all projects. By default this is only allowed by admin users. |
| deleted (Optional) | query | boolean | Show deleted items. 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 caution. It is typically only available to administrators. |
| ip6 (Optional) | query | string | <p>An IPv6 address to filter results by.</p> <p>New in version 2.5</p> |
| tags (Optional) | query | string | <p>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.</p> <p>New in version 2.26</p> |
| tags-any (Optional) | query | string | <p>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.</p> <p>New in version 2.26</p> |

| Name | In | Type | Description |
|-------------------------|-------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 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. New in version 2.26 |
| 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. New in version 2.26 |

Response

| Name | In | Type | Description |
|---------|------|--------|---------------------------------------------------------------------------------------------------------|
| servers | body | array | A list of <code>server</code> objects. |
| id | body | string | The UUID of the server. |
| links | body | array | Links to the resources in question. See API Guide / Links and References for more info. |
| name | body | string | The server name. |

Example List Servers: JSON response

```
{
  "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"
    }
  ]
}
```

POST

/servers

Create Server

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.

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)`, `entityTooLarge(413)`

Request

| Name | In | Type | Description |
|-----------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| server | body | object | A <code>server</code> object. |
| name | body | string | The server name. |
| flavorRef | body | string | The flavor reference, as a UUID or full URL, for the flavor for your server instance. |
| imageRef | 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. |

| Name | In | Type | Description |
|---------------------------------------------------|------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| security_groups (Optional) | body | array | One or more security groups. Specify the name of the security group in the <code>name</code> attribute. If you omit this attribute, the API creates the server in the <code>default</code> security group. |
| metadata (Optional) | body | object | Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each. |
| 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 | body | string | The administrative password of the server. |
| user_data (Optional) | body | string | Configuration information or scripts to use upon launch. Must be Base64 encoded. NOTE: The 'null' value allowed in Nova legacy v2 API, but due to the strict input validation, it isn't allowed in Nova v2.1 API. |
| os-availability-zone: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, you 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. |
| networks (Optional) | body | object | A <code>networks</code> object. Required parameter when there are multiple networks defined for the tenant. When you do not specify the <code>networks</code> 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 <code>uuid</code> attribute in a <code>networks</code> object. To provision the server instance with a NIC for an already existing port, specify the <code>port-id</code> in the <code>port</code> attribute in a <code>networks</code> object. Starting in microversion 2.32, it's possible to optionally assign an arbitrary tag to a virtual network interface, specify the <code>tag</code> attribute in the <code>network</code> object. An interface's tag is 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. |
| networks.uuid (Optional) | body | string | To provision the server instance with a NIC for a network, specify the UUID of the network in the <code>uuid</code> attribute in a <code>networks</code> object. Required if you omit the <code>port</code> attribute. |
| networks.port (Optional) | body | string | To provision the server instance with a NIC for an already existing port, specify the <code>port-id</code> in the <code>port</code> attribute in a <code>networks</code> object. The port status must be <code>DOWN</code> . required if you omit the <code>uuid</code> attribute. |
| networks.fixed_ip (Optional) | body | string | A fixed IPv4 address for the NIC. Valid with a <code>neutron</code> or <code>nova-networks</code> network. |
| networks.tag (Optional) | body | string | An arbitrary tag. |

| Name | In | Type | Description |
|-----------------------------------------------------|------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | New in version 2.32 |
| personality (Optional) | body | string | 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. |
| block_device_mapping_v2 (Optional) | body | object | <p>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:</p> <pre>"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" }</pre> <p>Starting in microversion 2.32, the tag is an optional, arbitrary 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.</p> |
| block_device_mapping_v2.device_name | body | string | A path to the device for the volume that you want to use to boot the server. |
| block_device_mapping_v2.source_type | body | string | The source type of the volume. A valid value is <code>blank</code> , <code>snapshot</code> , <code>volume</code> , or <code>image</code> . |
| block_device_mapping_v2.destination_type (Optional) | body | string | Defines where the volume comes from. A valid value is <code>local</code> or <code>volume</code> . |
| block_device_mapping_v2.delete_on_termination | body | string | To delete the boot volume when the server is destroyed, specify <code>true</code> . Otherwise, specify <code>false</code> . |
| block_device_mapping_v2.guest_format | body | string | Specifies the guest server disk file system format, such as <code>ephemeral</code> or <code>swap</code> . |
| block_device_mapping_v2.boot_index | body | string | 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 <code>NONE</code> . The simplest usage is, set the boot index of the boot device to 0 and use the default boot index value, <code>NONE</code> , 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.tag (Optional) | body | string | <p>An arbitrary tag.</p> <p>New in version 2.32</p> |
| config_drive (Optional) | body | boolean | Indicates whether a configuration drive enables metadata injection. The <code>config_drive</code> setting provides information about a drive that the instance can |

| Name | In | Type | Description |
|-------------------------------|------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | 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 <code>config_drive</code> . Read more in the OpenStack End User Guide. |
| key_name (Optional) | body | string | Key pair name. |
| os:scheduler_hints (Optional) | body | object | The dictionary of data to send to the scheduler. Alternatively, you can specify <code>OS-SCH-HNT:scheduler_hints</code> as the string. |
| OS-DCF:diskConfig (Optional) | body | string | Controls how the API partitions the disk when you create, rebuild, or resize servers. A server inherits the <code>OS-DCF:diskConfig</code> value from the image from which it was created, and an image inherits the <code>OS-DCF:diskConfig</code> 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 <code>OS-DCF:diskConfig</code> value for an image is <code>MANUAL</code> , you cannot create a server from that image and set its <code>OS-DCF:diskConfig</code> value to <code>AUTO</code> . A valid value is: <ul style="list-style-type: none"> <code>AUTO</code>. 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. <code>MANUAL</code>. 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. |
| 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 |

Example Create Server: JSON request

```
{
  "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
dCBtb3ZlcyBpbjBqdXN0IHN1Y2ggYSBkaXJlY3Rpb24gYW5kIGF0IHN1Y2ggYSBzcGVlZC4uLk10IGZlZWxzIGFuIGltcHVSc2lvbi4uLnRoXMGaXMGdGh1IHBSYWNlIHRvIGdvIG5vdy4gQnV0IHRoZSBza3kga25vd3MgdGh1IHJlYXNvbnMgYW5kIHRoZSBWYXR0ZXJucyBiZWphbmQgYWxsIGNsb3VkcyywY5kIHlvZSB3aWxsIGtub3csIHRvbywgd2hlbiB5b3UgbGlmZCB5b3VyZS5zZiBoawdoIGVub3VnaCB0byBZWUgYmV5b25kIGhvcml6b25zLiINCgOKLVJpY2hhcmQgQmFjaA=="
      }
    ],
    "security_groups": [
      {
        "name": "default"
      }
    ],
    "user_data" : "IyEvYmluL2Jhc2gKL2Jpbj9zdQply2hvICJJIGFtIGluIHlvdSEiCg=="
  },
  "OS-SCH-HNT:scheduler_hints": {
    "same_host": "48e6a9f6-30af-47e0-bc04-acaed113bb4e"
  }
}
```

Example Create Server (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"
    }]
  }
}
```

Response

| Name | In | Type | Description |
|-----------|------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| server | body | object | A server object. |
| addresses | body | object | The addresses for the server. Addresses information is hidden for any server in a state set in the <code>osapi_hide_server_address_states</code> configuration option. By default, servers in <code>building</code> state hide their addresses information. See <code>nova.conf</code> configuration options < http://docs.openstack.org/liberty/config-reference/content/list-of-compute-config-options.html >`_. |
| 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, <code>2015-08-27T09:49:58-05:00</code> . The <code>±hh:mm</code> value, if included, is the time zone as an offset from UTC. In the previous example, the offset value is <code>-05:00</code> . |
| 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. |
| hostId | body | string | The ID of the host. |
| id | body | string | The UUID of the server. |
| image | body | object | The UUID and links for the image for your server instance. The <code>image</code> 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. |

| Name | In | Type | Description |
|--------------------------------------|------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| metadata | body | object | A dictionary of metadata key-and-value pairs, which is maintained for backward compatibility. |
| name | body | string | The server name. |
| 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. |
| config_drive | body | boolean | Indicates whether or not a config drive was used for this server. |
| OS-DCF:diskConfig | body | string | Disk configuration. The value is either: <ul style="list-style-type: none"> 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 host name. Appears in the response for administrative users only. |
| OS-EXT-SRV-ATTR:hypervisor_hostname | body | string | The hypervisor host name. 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. |
| 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-SRV-USG:launched_at | body | string | <p>The date and time when the server was launched.</p> <p>The date and time stamp format is ISO 8601:</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>For example, 2015-08-27T09:49:58-05:00.</p> <p>The hh±:mm value, if included, is the time zone as an offset from UTC. If the <code>deleted_at</code> date and time stamp is not set,</p> |

| Name | In | Type | Description |
|---------------------------------|------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | its value is null. |
| OS-SRV-USG:terminated_at | body | string | <p>The date and time when the server was deleted.</p> <p>The date and time stamp format is ISO 8601:</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| progress | body | integer | A percentage value of the build progress. |
| security_groups | body | array | Security groups object. |
| security_group.name | body | string | The security group name. |
| status | body | string | The server status. |
| host_status | body | string | <p>The host status. Values where next value in list can override the previous:</p> <ul style="list-style-type: none"> • 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. |
| tenant_id | body | string | The UUID of the tenant in a multi-tenancy cloud. |
| updated | body | string | <p>The date and time when the resource was updated. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| user_id | body | string | The user ID of the user who owns the server. |
| OS-EXT-SERV-ATTR:hostname | body | string | <p>The hostname set on the instance when it is booted.</p> <p>New in version 2.3</p> |
| OS-EXT-SERV-ATTR:reservation_id | body | string | <p>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.</p> <p>New in version 2.3</p> |

| Name | In | Type | Description |
|-----------------------------------|------|--------|------------------------------------------------------------------------------------------------------------------------------------------|
| OS-EXT-SERV-ATTR:launch_index | body | int | When servers are launched via multiple create, this is the sequence in which the servers were launched. New in version 2.3 |
| OS-EXT-SERV-ATTR:kernel_id | body | string | The UUID of the kernel image when using an AMI. Will be null if not. New in version 2.3 |
| OS-EXT-SERV-ATTR:ramdisk_id | body | string | The UUID of the ramdisk image when using an AMI. Will be null if not. New in version 2.3 |
| OS-EXT-SERV-ATTR:root_device_name | body | string | The root device name for the instance New in version 2.3 |
| OS-EXT-SERV-ATTR:user_data | body | string | The user_data the instance was created with. New in version 2.3 |

Example Create Server: JSON response

```
{
  "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"
      }
    ]
  }
}
```

POST

/servers

Create Multiple Servers

There is a second kind of create call which can build multiple servers at once. This supports all the same parameters as create with a few additional attributes specific to multiple create.

Error handling for multiple create is not as consistent as for single server create, and there is no guarantee that all the servers will be built. This call should generally be avoided in favor of clients doing direct individual server creates.

Request (Additional Parameters)

These are the parameters beyond single create that are supported.

| Name | In | Type | Description |
|----------------------------------|------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| name | body | string | A base name for creating unique names during multiple create. A unique string will be appended to the end of this base name for every server created. |
| min_count (Optional) | body | string | The min number of servers to be created. Defaults to 1. |
| max_count (Optional) | body | string | The max number of servers to be created. Defaults to the value of min_count. |
| return_reservation_id (Optional) | body | boolean | Set to True to request that the response return a reservation ID instead of instance information. Default is False . |

Example Multiple Create with reservation ID

```
{
  "server": {
    "name": "new-server-test",
    "imageRef": "70a599e0-31e7-49b7-b260-868f441e862b",
    "flavorRef": "1",
    "metadata": {
      "My Server Name": "Apache1"
    },
    "return_reservation_id": "True",
    "min_count": "2",
    "max_count": "3"
  }
}
```

Response

| Name | In | Type | Description |
|----------------|------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| reservation_id | 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. |

If return_reservation_id is set to true only the reservation_id will be returned. This can be used as a filter with list servers detailed to see the status of all the servers being built.

Example Create multiple servers with reservation ID

```
{
  "reservation_id": "r-3fhpjulg"
}
```

If return_reservation_id is set to false a representation of the first server will be returned.

Example Create multiple servers without reservation ID

```
{
  "server": {
    "OS-DCF:diskConfig": "AUTO",
    "adminPass": "wfksh3GTTseP",
    "id": "440cf918-3ee0-4143-b289-f63e1d2000e6",
    "links": [
      {
```



```

    "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/servers/440cf918-3ee0-4143-b289-f63e1d2000e6",
    "rel": "self"
  },
  {
    "href": "http://openstack.example.com/6f70656e737461636b20342065766572/servers/440cf918-3ee0-4143-b289-f63e1d2000e6",
    "rel": "bookmark"
  }
],
"security_groups": [
  {
    "name": "default"
  }
]
}
}
}

```

GET

/servers/detail

List Servers Detailed

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.

To hide addresses information for instances in a certain state, set the `osapi_hide_server_address_states` configuration option. Set this option to a valid VM state in the `nova.conf` configuration file.

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 |
|---------------------|-------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| limit (Optional) | query | integer | Requests a page size of items. Returns a number of items up to a limit value. Use the <code>limit</code> parameter to make an initial limited request and use the ID of the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request. |
| marker (Optional) | query | string | The ID of the last-seen item. Use the <code>limit</code> parameter to make an initial limited request and use the ID of the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request. |
| sort_key (Optional) | query | string | Sorts by a server attribute. Default attribute is <code>created</code> . 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 server <code>sort_key</code> attribute. |
| sort_dir (Optional) | query | string | Sort direction. A valid value is <code>asc</code> (ascending) or <code>desc</code> (descending). Default is <code>desc</code> . 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 |

| Name | In | Type | Description |
|---------------------------|-------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | of the direction of the server <code>sort_key</code> attribute. |
| changes-since (Optional) | query | string | <p>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.</p> <p>The date and time stamp format is ISO 8601:</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>The ±hh:mm value, if included, returns the time zone as an offset from UTC. For example, 2015-08-27T09:49:58-05:00. If you omit the time zone, the UTC time zone is assumed.</p> |
| image (Optional) | query | string | Filters the response by an image, as a UUID. |
| flavor (Optional) | query | string | Filters the response by a flavor, as a UUID. A flavor is a combination of memory, disk size, and CPUs. |
| 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 <code>?name=bob</code> regular expression returns both bob and bobb. If you must match on only bob, you can use a regular expression that matches the syntax of the underlying database server that is implemented for Compute, such as MySQL or PostgreSQL. |
| status (Optional) | query | string | Filters the response by a server status, as a string. For example, ACTIVE. |
| ip (Optional) | query | string | An IPv4 address to filter results by. |
| reservation_id (Optional) | query | string | A reservation id as returned by a servers multiple create call. |
| all_tenants (Optional) | query | integer | Specify the <code>all_tenants=1</code> query parameter to list all instances for all projects. By default this is only allowed by admin users. |
| deleted (Optional) | query | boolean | Show deleted items. 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 caution. It is typically only available to administrators. |
| ip6 (Optional) | query | string | <p>An IPv6 address to filter results by.</p> <p>New in version 2.5</p> |
| tags (Optional) | query | string | <p>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.</p> <p>New in version 2.26</p> |
| 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 |

| Name | In | Type | Description |
|-------------------------|-------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | is 't1 OR t2'. Tags in query must be separated by comma. New in version 2.26 |
| 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. New in version 2.26 |
| 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. New in version 2.26 |

Response

| Name | In | Type | Description |
|-----------|------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| server | body | object | A server object. |
| addresses | body | object | The addresses for the server. Addresses information is hidden for any server in a state set in the <code>osapi_hide_server_address_states</code> configuration option. By default, servers in <code>building</code> state hide their addresses information. See <code>nova.conf</code> configuration options <http://docs.openstack.org/liberty/config-reference/content/list-of-compute-config-options.html> `. |
| 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 | The ID and links for the flavor for your server instance. A flavor is a combination of memory, disk size, and CPUs. |
| hostId | body | string | The ID of the host. |
| id | body | string | The UUID of the server. |
| image | body | object | The UUID and links for the image for your server instance. The <code>image</code> 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. |

| Name | In | Type | Description |
|--------------------------------------|------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| metadata | body | object | A dictionary of metadata key-and-value pairs, which is maintained for backward compatibility. |
| name | body | string | The server name. |
| 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. |
| config_drive | body | boolean | Indicates whether or not a config drive was used for this server. |
| OS-DCF:diskConfig | body | string | Disk configuration. The value is either: <ul style="list-style-type: none"> 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 host name. Appears in the response for administrative users only. |
| OS-EXT-SRV-ATTR:hypervisor_hostname | body | string | The hypervisor host name. 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. |
| 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-SRV-USG:launched_at | body | string | <p>The date and time when the server was launched.</p> <p>The date and time stamp format is ISO 8601:</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>For example, 2015-08-27T09:49:58-05:00.</p> <p>The hh±:mm value, if included, is the time zone as an offset from UTC. If the <code>deleted_at</code> date and time stamp is not set, its value is <code>null</code>.</p> |

| Name | In | Type | Description |
|---------------------------------|------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OS-SRV-USG:terminated_at | body | string | <p>The date and time when the server was deleted.</p> <p>The date and time stamp format is ISO 8601:</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| progress | body | integer | A percentage value of the build progress. |
| security_groups | body | array | Security groups object. |
| security_group.name | body | string | The security group name. |
| status | body | string | The server status. |
| host_status | body | string | <p>The host status. Values where next value in list can override the previous:</p> <ul style="list-style-type: none"> • 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. |
| tenant_id | body | string | The UUID of the tenant in a multi-tenancy cloud. |
| updated | body | string | <p>The date and time when the resource was updated. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| user_id | body | string | The user ID of the user who owns the server. |
| OS-EXT-SERV-ATTR:hostname | body | string | <p>The hostname set on the instance when it is booted.</p> <p>New in version 2.3</p> |
| OS-EXT-SERV-ATTR:reservation_id | body | string | <p>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.</p> <p>New in version 2.3</p> |
| OS-EXT-SERV-ATTR:launch_index | body | int | When servers are launched via multiple create, this is the sequence in which the servers were launched. |

| Name | In | Type | Description |
|-----------------------------------|------|--------|--------------------------------------------------------------------------------------------------------|
| | | | New in version 2.3 |
| OS-EXT-SERV-ATTR:kernel_id | body | string | The UUID of the kernel image when using an AMI. Will be null if not. New in version 2.3 |
| OS-EXT-SERV-ATTR:ramdisk_id | body | string | The UUID of the ramdisk image when using an AMI. Will be null if not. New in version 2.3 |
| OS-EXT-SERV-ATTR:root_device_name | body | string | The root device name for the instance New in version 2.3 |
| OS-EXT-SERV-ATTR:user_data | body | string | The user_data the instance was created with. New in version 2.3 |

Example List Servers Detailed: JSON response

```
{
  "servers": [
    {
      "accessIPv4": "1.2.3.4",
      "accessIPv6": "80fe::",
      "addresses": {
        "private": [
          {
            "addr": "192.168.0.3",
            "OS-EXT-IPS-MAC:mac_addr": "aa:bb:cc:dd:ee:ff",
            "OS-EXT-IPS:type": "fixed",
            "version": 4
          }
        ]
      },
      "created": "2013-09-03T04:01:32Z",
      "flavor": {
        "id": "1",
        "links": [
          {
            "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/1",
            "rel": "bookmark"
          }
        ]
      },
      "hostId": "bcf92836fc9ed4203a75cb0337afc7f917d2be504164b995c2334b25",
      "id": "f5dc173b-6804-445a-a6d8-c705dad5b5eb",
      "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/v2/6f70656e737461636b20342065766572/servers/f5dc173b-6804-445a-a6d8-c705dad5b5eb",
          "rel": "self"
        },
        {
          "href": "http://openstack.example.com/6f70656e737461636b20342065766572/servers/f5dc173b-6804-445a-a6d8-c705dad5b5eb",
          "rel": "bookmark"
        }
      ],
      "metadata": {
        "My Server Name": "Apache1"
      },
      "name": "new-server-test",
      "config_drive": "",
      "OS-DCF:diskConfig": "AUTO",
      "OS-EXT-AZ:availability_zone": "nova",
      "OS-EXT-SRV-ATTR:host": "c3f14e9812ad496baf92ccfb3c61e15f",
      "OS-EXT-SRV-ATTR:hypervisor_hostname": "fake-mini",
      "OS-EXT-SRV-ATTR:instance_name": "instance-00000001",
      "OS-EXT-STS:power_state": 1,
      "OS-EXT-STS:task_state": null,
      "OS-EXT-STS:vm_state": "active",
      "os-extended-volumes:volumes_attached": [
        {
          "id": "volume_id1"
        },
        {
          "id": "volume_id2"
        }
      ]
    }
  ]
}
```

```

    ],
    "OS-SRV-USG:launched_at": "2013-09-23T13:53:12.774549",
    "OS-SRV-USG:terminated_at": null,
    "progress": 0,
    "security_groups": [
      {
        "name": "default"
      }
    ],
    "status": "ACTIVE",
    "tenant_id": "6f70656e737461636b20342065766572",
    "updated": "2013-09-03T04:01:32Z",
    "user_id": "fake"
  }
]
}

```

GET

/servers/{server_id}

Show Server Details

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.

To hide addresses information for instances in a certain state, set the `osapi_hide_server_address_states` configuration option. Set this option to a valid VM state in the `nova.conf` configuration file.

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. |
| addresses | body | object | The addresses for the server. Addresses information is hidden for any server in a state set in the <code>osapi_hide_server_address_states</code> configuration option. By default, servers in <code>building</code> state hide their addresses information. See <code>nova.conf</code> configuration options |

| Name | In | Type | Description |
|-----------------------------|------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | < http://docs.openstack.org/liberty/config-reference/content/list-of-compute-config-options.html >`_. |
| created | body | string | <p>The date and time when the resource was created. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| 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. |
| hostId | body | string | The ID of the host. |
| 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. |
| 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. |
| config_drive | body | boolean | Indicates whether or not a config drive was used for this server. |
| OS-DCF:diskConfig | body | string | <p>Disk configuration. The value is either:</p> <ul style="list-style-type: none"> 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. |

| Name | In | Type | Description |
|--------------------------------------|------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OS-EXT-SRV-ATTR:host | body | string | The host name. Appears in the response for administrative users only. |
| OS-EXT-SRV-ATTR:hypervisor_hostname | body | string | The hypervisor host name. 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. |
| 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-SRV-USG:launched_at | body | string | <p>The date and time when the server was launched.</p> <p>The date and time stamp format is ISO 8601:</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>For example, 2015-08-27T09:49:58-05:00.</p> <p>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.</p> |
| OS-SRV-USG:terminated_at | body | string | <p>The date and time when the server was deleted.</p> <p>The date and time stamp format is ISO 8601:</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| progress | body | integer | A percentage value of the build progress. |
| security_groups | body | array | Security groups object. |
| security_group.name | body | string | The security group name. |
| status | body | string | The server status. |
| host_status | body | string | <p>The host status. Values where next value in list can override the previous:</p> <ul style="list-style-type: none"> UP if nova-compute up. UNKNOWN if nova-compute not reported by servicegroup |

| Name | In | Type | Description |
|-----------------------------------|------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | <p>driver.</p> <ul style="list-style-type: none"> 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. |
| tenant_id | body | string | The UUID of the tenant in a multi-tenancy cloud. |
| updated | body | string | <p>The date and time when the resource was updated. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| user_id | body | string | The user ID of the user who owns the server. |
| fault (Optional) | body | object | A fault object. Only displayed in the failed response. Default keys are code, created, and message (response code, created time, and message respectively). In addition, the key details (stack trace) is available if you have the administrator privilege. |
| OS-EXT-SERV-ATTR:hostname | body | string | <p>The hostname set on the instance when it is booted.</p> <p>New in version 2.3</p> |
| OS-EXT-SERV-ATTR:reservation_id | body | string | <p>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.</p> <p>New in version 2.3</p> |
| OS-EXT-SERV-ATTR:launch_index | body | int | <p>When servers are launched via multiple create, this is the sequence in which the servers were launched.</p> <p>New in version 2.3</p> |
| OS-EXT-SERV-ATTR:kernel_id | body | string | <p>The UUID of the kernel image when using an AMI. Will be null if not.</p> <p>New in version 2.3</p> |
| OS-EXT-SERV-ATTR:ramdisk_id | body | string | <p>The UUID of the ramdisk image when using an AMI. Will be null if not.</p> <p>New in version 2.3</p> |
| OS-EXT-SERV-ATTR:root_device_name | body | string | <p>The root device name for the instance</p> <p>New in version 2.3</p> |
| OS-EXT-SERV- | body | string | The user_data the instance was created with. |

| Name | In | Type | Description |
|----------------|----|------|--------------------|
| ATTR:user_data | | | New in version 2.3 |

Example Show Server Details: JSON response

```
{
  "server": {
    "accessIPv4": "1.2.3.4",
    "accessIPv6": "80fe::",
    "addresses": {
      "private": [
        {
          "addr": "192.168.0.3",
          "OS-EXT-IPS-MAC:mac_addr": "aa:bb:cc:dd:ee:ff",
          "OS-EXT-IPS:type": "fixed",
          "version": 4
        }
      ]
    },
    "created": "2013-09-03T04:01:32Z",
    "flavor": {
      "id": "1",
      "links": [
        {
          "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/1",
          "rel": "bookmark"
        }
      ]
    },
    "hostId": "92154fab69d5883ba2c8622b7e65f745dd33257221c07af363c51b29",
    "id": "0e44cc9c-e052-415d-afbf-469b0d384170",
    "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/v2/6f70656e737461636b20342065766572/servers/0e44cc9c-e052-415d-afbf-469b0d384170",
        "rel": "self"
      },
      {
        "href": "http://openstack.example.com/6f70656e737461636b20342065766572/servers/0e44cc9c-e052-415d-afbf-469b0d384170",
        "rel": "bookmark"
      }
    ],
    "metadata": {
      "My Server Name": "Apache1"
    },
    "name": "new-server-test",
    "config_drive": "",
    "OS-DCF:diskConfig": "AUTO",
    "OS-EXT-AZ:availability_zone": "nova",
    "OS-EXT-SRV-ATTR:host": "b8b357f7100d4391828f2177c922ef93",
    "OS-EXT-SRV-ATTR:hypervisor_hostname": "fake-mini",
    "OS-EXT-SRV-ATTR:instance_name": "instance-00000001",
    "OS-EXT-STS:power_state": 1,
    "OS-EXT-STS:task_state": null,
    "OS-EXT-STS:vm_state": "active",
    "os-extended-volumes:volumes_attached": [
      {
        "id": "volume_id1"
      },
      {
        "id": "volume_id2"
      }
    ],
    "OS-SRV-USG:launched_at": "2013-09-23T13:37:00.880302",
    "OS-SRV-USG:terminated_at": null,
    "progress": 0,
    "security_groups": [
      {
        "name": "default"
      }
    ],
    "status": "ACTIVE",
    "tenant_id": "6f70656e737461636b20342065766572",
    "updated": "2013-09-03T04:01:33Z",
    "user_id": "fake"
  }
}
```

PUT
`/servers/{server_id}`

Update Server

Updates the editable attributes of an existing server.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|------------------------------|------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| server_id | path | string | The UUID of the server. |
| 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. |
| name | body | string | The server name. |
| OS-DCF:diskConfig (Optional) | body | string | <p>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:</p> <ul style="list-style-type: none">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. |
| description (Optional) | body | string | <p>A free form description of the server. Limited to 255 characters in length. Before microversion 2.19 this was set to the server name.</p> <p>New in version 2.19</p> |

Example Update server name: JSON request

```
{
  "server": {
    "accessIPv4": "4.3.2.1",
    "accessIPv6": "80fe::",
    "OS-DCF:diskConfig": "AUTO",
    "name" : "new-server-test"
  }
}
```

Response

| Name | In | Type | Description |
|------------------------------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| server | body | object | A server object. |
| addresses | body | object | The addresses for the server. Addresses information is hidden for any server in a state set in the <code>osapi_hide_server_address_states</code> configuration option. By default, servers in <code>building</code> state hide their addresses information. See <code>nova.conf</code> configuration options http://docs.openstack.org/liberty/config-reference/content/list-of-compute-config-options.html > ` _. |
| created | body | string | <p>The date and time when the resource was created. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| 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. |
| hostId | body | string | The ID of the host. |
| id | body | string | The UUID of the server. |
| image | body | object | The UUID and links for the image for your server instance. The <code>image</code> object might be an empty string when you boot the server from a volume. |
| key_name (Optional) | body | string | Key pair name. |
| links | body | array | Links to the resources in question. See API Guide / Links and References for more info. |
| metadata (Optional) | body | object | Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each. |
| name | body | string | The server name. |
| 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. |
| OS-DCF:diskConfig (Optional) | body | string | <p>Controls how the API partitions the disk when you create, rebuild, or resize servers. A server inherits the <code>OS-DCF:diskConfig</code> value from the image from which it was created, and an image inherits the <code>OS-DCF:diskConfig</code> 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 <code>OS-DCF:diskConfig</code> value for an image is <code>MANUAL</code>, you cannot create a server from that image and set its <code>OS-DCF:diskConfig</code> value to <code>AUTO</code>. A valid value is:</p> <ul style="list-style-type: none"> 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 |

| Name | In | Type | Description |
|--------------------------------------|------|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | 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 host name. Appears in the response for administrative users only. |
| OS-EXT-SRV-ATTR:hypervisor_hostname | body | string | The hypervisor host name. 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. |
| 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-SRV-USG:launched_at | body | string | <p>The date and time when the server was launched.</p> <p>The date and time stamp format is ISO 8601:</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>For example, 2015-08-27T09:49:58-05:00.</p> <p>The hh±mm value, if included, is the time zone as an offset from UTC. If the <code>deleted_at</code> date and time stamp is not set, its value is <code>null</code>.</p> |
| OS-SRV-USG:terminated_at | body | string | <p>The date and time when the server was deleted.</p> <p>The date and time stamp format is ISO 8601:</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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 <code>deleted_at</code> date and time stamp is not set, its value is <code>null</code>.</p> |
| progress | body | integer | A percentage value of the build progress. |
| security_groups | body | array | Security groups object. |
| security_group.name | body | string | The security group name. |
| status | body | string | The server status. |
| host_status | body | string | <p>The host status. Values where next value in list can override the previous:</p> <ul style="list-style-type: none"> 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. |

| Name | In | Type | Description |
|-----------------------------------|------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | <ul style="list-style-type: none"> Empty string indicates there is no host for server. This attribute appears in the response only if the policy permits. |
| tenant_id | body | string | The UUID of the tenant in a multi-tenancy cloud. |
| updated | body | string | <p>The date and time when the resource was updated. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| user_id | body | string | The user ID of the user who owns the server. |
| OS-EXT-SERV-ATTR:hostname | body | string | <p>The hostname set on the instance when it is booted.</p> <p>New in version 2.3</p> |
| OS-EXT-SERV-ATTR:reservation_id | body | string | <p>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.</p> <p>New in version 2.3</p> |
| OS-EXT-SERV-ATTR:launch_index | body | int | <p>When servers are launched via multiple create, this is the sequence in which the servers were launched.</p> <p>New in version 2.3</p> |
| OS-EXT-SERV-ATTR:kernel_id | body | string | <p>The UUID of the kernel image when using an AMI. Will be null if not.</p> <p>New in version 2.3</p> |
| OS-EXT-SERV-ATTR:ramdisk_id | body | string | <p>The UUID of the ramdisk image when using an AMI. Will be null if not.</p> <p>New in version 2.3</p> |
| OS-EXT-SERV-ATTR:root_device_name | body | string | <p>The root device name for the instance</p> <p>New in version 2.3</p> |
| OS-EXT-SERV-ATTR:user_data | body | string | <p>The user_data the instance was created with.</p> <p>New in version 2.3</p> |

Example Update server name: JSON response

```
{
  "server": {
    "OS-DCF:diskConfig": "AUTO",
    "accessIPv4": "1.2.3.4",
    "accessIPv6": "80fe::",
    "addresses": {
      "private": [
        {
          "addr": "192.168.0.3",
          "version": 4
        }
      ]
    },
    "created": "2012-12-02T02:11:57Z",
    "flavor": {
```

```

    "id": "1",
    "links": [
      {
        "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/1",
        "rel": "bookmark"
      }
    ]
  },
  "hostId": "6e84af987b4e7ec1c039b16d21f508f4a505672bd94fb0218b668d07",
  "id": "324dfb7d-f4a9-419a-9a19-237df04b443b",
  "image": {
    "id": "70a599e0-31e7-49b7-b260-868f441e862b",
    "links": [
      {
        "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/70a599e0-31e7-49b7-b260-868f441e862b",
        "rel": "bookmark"
      }
    ]
  },
  "links": [
    {
      "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/servers/324dfb7d-f4a9-419a-9a19-237df04b443b",
      "rel": "self"
    },
    {
      "href": "http://openstack.example.com/6f70656e737461636b20342065766572/servers/324dfb7d-f4a9-419a-9a19-237df04b443b",
      "rel": "bookmark"
    }
  ],
  "metadata": {
    "My Server Name": "Apache1"
  },
  "name": "new-server-test",
  "progress": 0,
  "status": "ACTIVE",
  "tenant_id": "6f70656e737461636b20342065766572",
  "updated": "2012-12-02T02:11:58Z",
  "user_id": "fake"
}
}

```

DELETE

/servers/{server_id}

Delete Server

Deletes a server.

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 **DELETED** through API calls.
- The port attached to the server is deleted.
- The server does not appear in the list servers response.
- 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

Servers - run an action (servers, action)

Enables all users to perform an action on a server. Specify the action in the request body.

You can associate a fixed or floating IP address with a server, or disassociate a fixed or floating IP address from a server. You can attach a volume to a server.

You can create an image from a server, evacuate a server from a failed host to a new host, and force-delete a server before deferred cleanup. You can lock, pause, reboot, rebuild, rescue, resize, resume, confirm the resize of, revert a pending resize for, shelve, shelf-offload, unshelve, start, stop, unlock, unpause, and unrescue a server. You can also change the password of the server and add a security group to or remove a security group from a server. You can also trigger a crash dump into a server since Mitaka release.

You can get an RDP, serial, SPICE, or VNC console for a server.

POST

/servers/{server_id}/action

Add (Associate) Floating Ip (addFloatingIp Action)

Adds a floating IP address to a server, which associates that address with the server.

A pool of floating IP addresses, configured by the cloud administrator, is available in OpenStack Compute. The project quota defines the maximum number of floating IP addresses that you can allocate to the project. After you [create \(allocate\) a floating IP address](#) for a project, you can associate that address with the server. Specify the `addFloatingIp` action in the request body.

If an instance is connected to multiple networks, you can associate a floating IP address with a specific fixed IP address by using the optional `fixed_address` parameter.

Preconditions

The server must exist.

You can only add a floating IP address to the server when its status is `available`.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|--------------------------|------|--------|--------------------------------------------------------------------------------|
| server_id | path | string | The UUID of the server. |
| addFloatingIp | body | string | The action. |
| address | body | string | The floating IP address. |
| fixed_address (Optional) | body | string | The fixed IP address with which you want to associate the floating IP address. |

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Add Security Group To A Server (addSecurityGroup Action)

Adds a security group to a server.

Specify the addSecurityGroup action in the request body.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|------------------|------|--------|--------------------------|
| server_id | path | string | The UUID of the server. |
| addSecurityGroup | body | string | The action. |
| name | body | string | The security group name. |

Example Add Security Group To A Server (addSecurityGroup Action): JSON request

```
{
  "addSecurityGroup": {
    "name": "test"
  }
}
```

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Change Administrative Password (changePassword Action)

Changes the administrative password for a server.

Specify the `changePassword` action in the request body.

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

Normal response codes: 202

Error response codes: `badRequest(400)`, `unauthorized(401)`, `forbidden(403)`, `itemNotFound(404)`, `conflict(409)`, `notImplemented(501)`

Request

| Name | In | Type | Description |
|----------------|------|--------|-------------------------|
| server_id | path | string | The UUID of the server. |
| changePassword | body | string | The action. |

Example Change Administrative Password (changePassword Action): JSON request

```
{
  "changePassword" : {
    "adminPass" : "foo"
  }
}
```

Response

There is no body content for the response of a successful POST operation.

POST

`/servers/{server_id}/action`

Confirm Resized Server (confirmResize Action)

Confirms a pending resize action for a server.

Specify the `confirmResize` action in the request body.

After you make this request, you typically must keep polling the server status to determine whether the request succeeded. A successfully confirming resize operation shows a status of `ACTIVE` or `SHUTOFF` and a `migration_status` of `confirmed`. You can also see the resized server in the compute node that OpenStack Compute manages.

Preconditions

You can only confirm the resized server where the status is `VERIFY_RESIZED`, the `vm_status` is `RESIZED`, and the `migration_status` is `finished` or `confirming`.

If the server is locked, you must have administrator privileges to confirm the server.

Troubleshooting

If the server status remains **RESIZED**, the request failed. Ensure you meet the preconditions and run the request again. If the request fails again, investigate the compute back end or ask your cloud provider.

Normal response codes: 204

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

Request

| Name | In | Type | Description |
|---------------|------|--------|-------------------------|
| server_id | path | string | The UUID of the server. |
| confirmResize | body | string | The action. |

Example Confirm Resized Server (confirmResize Action): JSON request

```
{
  "confirmResize" : null
}
```

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Create Image (createImage Action)

Creates an image from a server.

Specify the `createImage` action in the request body.

After you make this request, you typically must keep polling the status of the created image to determine whether the request succeeded.

If the operation succeeds, the created image has a status of `active` and the server status returns to the original status. You can also see the new image in the image back end that OpenStack Image service manages.

Preconditions

The server must exist.

You can only create a new image from the server when its status is **ACTIVE**, **SHUTOFF**, **PAUSED**, or **SUSPENDED**.

The connection to the Image service is valid.

Troubleshooting

If the image status remains uploading or shows another error status, the request failed. Ensure you meet the preconditions and run the request again. If the request fails again, investigate the image back end.

If the server status does not go back to an original server's status, the request failed. Ensure you meet the preconditions, or check if there is another operation that causes race conditions for the server, then run the request again. If the request fails again, investigate the compute back end or ask your cloud provider.

If the request fails due to an error on OpenStack Compute service, the image is purged from the image store that OpenStack Image service manages. Ensure you meet the preconditions and run the request again. If the request fails again, investigate OpenStack Compute service or ask your cloud provider.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|-------------|------|--------|-------------------------|
| server_id | path | string | The UUID of the server. |
| createImage | body | string | The action. |

Example Create Image (createImage Action): JSON request

```
{
  "createImage" : {
    "name" : "foo-image",
    "metadata": {
      "meta_var": "meta_val"
    }
  }
}
```

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Lock Server (lock Action)

Locks a server.

Specify the `lock` action in the request body.

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

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------|
| server_id | path | string | The UUID of the server. |
| lock | body | string | The action. |

Example Lock Server (lock Action): JSON request

```
{
  "lock": null
}
```

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Pause Server (pause Action)

Pauses a server. Changes its status to **PAUSED**.

Specify the **pause** action in the request body.

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

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. |
| pause | body | string | The action. |

Example Pause Server (pause Action): JSON request

```
{
  "pause": null
}
```

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Reboot Server (reboot Action)

Reboots a server.

Specify the `reboot` action in the request body.

Normal response codes: 202

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. |
| reboot | body | string | The action. |

Example Reboot Server (reboot Action): JSON request

```
{
  "reboot" : {
    "type" : "HARD"
  }
}
```

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Rebuild Server (rebuild Action)

Rebuilds a server.

Specify the `rebuild` action in the request body.

To rebuild the server with preservation of the ephemeral partition, set the `preserve_ephemeral` parameter to `true`.

Normal response codes: 202

Error response codes: badRequest(400), unauthorized(401), forbidden(403), itemNotFound(404), conflict(409), payloadTooLarge(413)

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------|
| server_id | path | string | The UUID of the server. |
| rebuild | body | string | The action. |

| Name | In | Type | Description |
|-------------------------------|------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| imageRef | 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. |
| name | body | string | The security group name. |
| adminPass (Optional) | body | string | An administrative password to access the evacuated or rebuilt instance. To set the administrative password, set the <code>enable_instance_password</code> configuration option to <code>True</code> . If you set this option to <code>False</code> and you try to set the administrative password, the API does not set the password and the response shows a <code>null</code> value for the <code>adminPass</code> parameter. Deprecated in version 2.14 |
| metadata (Optional) | body | object | Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each. |
| personality (Optional) | body | string | 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. |
| preserve_ephemeral (Optional) | body | boolean | Indicates whether the server is rebuilt with the preservation of the ephemeral partition (<code>true</code>). Note This only works with baremetal servers provided by Ironic. Passing it to any other server instance results in a fault and will prevent the rebuild from happening. |

Example Rebuild Server (rebuild Action): JSON request

Preserving the ephemeral disk

```
{
  "rebuild" : {
    "imageRef" : "70a599e0-31e7-49b7-b260-868f441e862b",
    "name" : "foobar",
    "adminPass" : "seekr3t",
    "metadata" : {
      "meta_var" : "meta_val"
    },
    "personality" : [
      {
        "path" : "/etc/banner.txt",
        "contents" : "ICAgICAgDQoiQSBjbG91ZCBkb2VzIG5vdCBrbm93IHdoeSBp
dCBtb3ZlcyBpbjBqdXN0IHN1Y2ggYSBkaXJlY3Rpb24gYW5kIGF0IHN1Y2ggYSBzcGVlZC4uLk10IGZlZWxzIGFuIGltchVs
c2lvbi4uLnRoXmgaXGdGh1IHBSYWNlIHhvIGdvIG5vdy4gQnV0IHROZSBza3kga25vd3MgdGh1IHJlYXNvbnMgYW5kIHRO
ZSBwYXR0ZXJucyBiZWphbmQgYWxsIGNsb3VkcycwYW5kIHlv dSB3awxsIGtub3csIHRvbywgd2hlbiB5b3UgbG1mdCB5b3Vy
c2VsZiBoaWdoIGVub3VnaCB0byBzZWUgYmV5b25kIGhvcml6 b25zLiINCg0KLVPjY2hhcmQgQmFjaA=="
      }
    ],
    "preserve_ephemeral": true
  }
}
```

Response

Example Rebuild Server (rebuild Action): JSON response

```
{
  "server": {
    "accessIPv4": "1.2.3.4",
```



```

"accessIPv6": "80fe::",
"OS-DCF:diskConfig": "AUTO",
"addresses": {
  "private": [
    {
      "addr": "192.168.0.3",
      "version": 4
    }
  ]
},
"adminPass": "seekr3t",
"created": "2013-11-14T06:29:00Z",
"flavor": {
  "id": "1",
  "links": [
    {
      "href": "http://openstack.example.com/6f70656e737461636b20342065766572/flavors/1",
      "rel": "bookmark"
    }
  ]
},
"hostId": "28d8d56f0e3a77e20891f455721cbb68032e017045e20aa5dfc6cb66",
"id": "a0a80a94-3d81-4a10-822a-daa0cf9e870b",
"image": {
  "id": "70a599e0-31e7-49b7-b260-868f441e862b",
  "links": [
    {
      "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/70a599e0-31e7-49b7-b260-868f441e862b",
      "rel": "bookmark"
    }
  ]
},
"links": [
  {
    "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/servers/a0a80a94-3d81-4a10-822a-daa0cf9e870b",
    "rel": "self"
  },
  {
    "href": "http://openstack.example.com/6f70656e737461636b20342065766572/servers/a0a80a94-3d81-4a10-822a-daa0cf9e870b",
    "rel": "bookmark"
  }
],
"metadata": {
  "meta_var": "meta_val"
},
"name": "foobar",
"progress": 0,
"status": "ACTIVE",
"tenant_id": "6f70656e737461636b20342065766572",
"updated": "2013-11-14T06:29:02Z",
"user_id": "fake"
}
}

```

POST

/servers/{server_id}/action

Remove (Disassociate) Floating Ip (removeFloatingIp Action)

Removes, or disassociates, a floating IP address from a server.

The IP address is returned to the pool of IP addresses that is available for all projects. When you remove a floating IP address and that IP address is still associated with a running instance, it is automatically disassociated from that instance.

Specify the `removeFloatingIp` action in the request body.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|------------------|------|--------|--------------------------|
| server_id | path | string | The UUID of the server. |
| removeFloatingIp | body | string | The action. |
| address | body | string | The floating IP address. |

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Remove Security Group From A Server (removeSecurityGroup Action)

Removes a security group from a server.

Specify the `removeSecurityGroup` action in the request body.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|---------------------|------|--------|--------------------------|
| server_id | path | string | The UUID of the server. |
| removeSecurityGroup | body | string | The action. |
| name | body | string | The security group name. |

Example Remove Security Group From A Server (removeSecurityGroup Action): JSON request

```
{
  "removeSecurityGroup": {
    "name": "test"
  }
}
```

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Rescue Server (rescue Action)

Puts a server in rescue mode and changes its status to RESCUE.

Specify the `rescue` action in the request body.

If you specify the `rescue_image_ref` extended attribute, the image is used to rescue the instance. If you omit an image reference, the base image reference is used by default.

Asynchronous Postconditions

After you successfully rescue a server and make a GET `/servers/{server_id}` request, its status changes to RESCUE.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|-----------------------------|------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| server_id | path | string | The UUID of the server. |
| rescue | body | string | The action. |
| adminPass (Optional) | body | string | The password for the rescued instance. If you omit this parameter, the operation generates a new password. |
| rescue_image_ref (Optional) | body | string | The image reference to use to rescue your server instance. Specify the image reference by ID or full URL. If you omit an image reference, default is the base image reference. |

Example Rescue server: JSON request

```
{
  "rescue": {
    "adminPass": "MySecretPass"
  }
}
```

Response

| Name | In | Type | Description |
|-----------|------|--------|--------------------------------------------|
| adminPass | body | string | The administrative password of the server. |

Example Extended rescue server: JSON response

```
{
  "adminPass": "MySecretPass"
}
```

POST

/servers/{server_id}/action

Resize Server (resize Action)

Resizes a server.

Specify the `resize` action in the request body.

A successfully resized server shows a `VERIFY_RESIZE` status, `RESIZED` VM status, and `finished` migration status. If you set the `resize_confirm_window` option of the Compute service to an integer value, the Compute service automatically confirms the resize operation after the set interval in

seconds.

Preconditions

You can only resize a server when its status is **ACTIVE** or **SHUTOFF**.

If the server is locked, you must have administrator privileges to resize the server.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------|
| server_id | path | string | The UUID of the server. |
| resize | body | string | The action. |

Example Resize Server (Resize Action): JSON request

```
{
  "resize" : {
    "flavorRef" : "2",
    "OS-DCF:diskConfig": "AUTO"
  }
}
```

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Resume Suspended Server (resume Action)

Resumes a suspended server and changes its status to **ACTIVE**.

Specify the `resume` action in the request body.

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

Normal response codes: 202

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. |

| Name | In | Type | Description |
|--------|------|--------|-------------|
| | | | |
| resume | body | string | The action. |

Example Resume Suspended Server (Resume Action): JSON request

```
{
  "resume": null
}
```

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Revert Resized Server (revertResize Action)

Cancels and reverts a pending resize action for a server.

Specify the `revertResize` action in the request body.

Preconditions

You can only revert the resized server where the status is `VERIFY_RESIZE` and the `OS-EXT-STS:vm_state` is `resized`.

If the server is locked, you must have administrator privileges to revert the resizing.

Asynchronous Postconditions

After you make this request, you typically must keep polling the server status to determine whether the request succeeded. A reverting resize operation shows a status of `REVERT_RESIZE` and a `task_state` of `resize_reverting`. If successful, the status will return to `ACTIVE` or `SHUTOFF`. You can also see the reverted server in the compute node that OpenStack Compute manages.

Troubleshooting

If the server status remains `VERIFY_RESIZE`, the request failed. Ensure you meet the preconditions and run the request again. If the request fails again, investigate the compute back end.

The server is not reverted in the compute node that OpenStack Compute manages.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|--------------|------|--------|-------------------------|
| server_id | path | string | The UUID of the server. |
| revertResize | body | string | The action. |

Example Revert Resized Server (revertResize Action): JSON request

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

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Start Server (os-start Action)

Starts a stopped server and changes its status to **ACTIVE**.

Specify the `os-start` action in the request body.

Preconditions

The server status must be **SHUTOFF**.

If the server is locked, you must have administrator privileges to start the server.

Asynchronous Postconditions

After you successfully start a server, its status changes to **ACTIVE**. The server appears on the compute node that the Compute service manages.

Troubleshooting

If the server status does not change to **ACTIVE**, the start operation failed. Ensure that you meet the preconditions and run the request again. If the request fails again, investigate whether another operation is running that causes a race condition.

Normal response codes: 202

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. |
| os-start | body | string | The action. |

Example Start server: JSON request

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

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Stop Server (os-stop Action)

Stops a running server and changes its status to SHUTOFF.

Specify the `os-stop` action in the request body.

Preconditions

The server status must be `ACTIVE` or `ERROR`.

If the server is locked, you must have administrator privileges to stop the server.

Asynchronous Postconditions

After you successfully stop a server, its status changes to `SHUTOFF`. The server instance data appears only on the compute node that Compute service manages.

Normal response codes: 202

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. |
| os-stop | body | string | The action. |

Example Stop server: JSON request

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

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Suspend Server (suspend Action)

Suspends a server and changes its status to **SUSPENDED**.

Specify the **suspend** action in the request body.

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

Normal response codes: 202

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. |
| suspend | body | string | The action. |

Example Suspend Server (suspend Action): JSON request

```
{
  "suspend": null
}
```

Response

There is no body content for the response of a successful POST operation.

POST

`/servers/{server_id}/action`

Unlock Server (unlock Action)

Unlocks a locked server.

Specify the **unlock** action in the request body.

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

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------|
| server_id | path | string | The UUID of the server. |
| unlock | body | string | The action. |

Example Unlock Server (unlock Action): JSON request

```
{
  "unlock": null
}
```

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Unpause Server (unpause Action)

Unpauses a paused server and changes its status to **ACTIVE**.

Specify the **unpause** action in the request body.

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

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. |
| unpause | body | string | The action. |

Example Unpause Server (unpause Action): JSON request

```
{
  "unpause": null
}
```

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Unrescue Server (unrescue Action)

Unrescues a server. Changes status to **ACTIVE**.

Specify the **unrescue** action in the request body.

Preconditions

The server must exist.

You can only unrescue a server when its status is **RESCUE**.

Asynchronous Postconditions

After you successfully unrescue a server and make a **GET /servers/{server_id}** request, its status changes to **ACTIVE**.

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. |
| unrescue | body | string | The action. |

Example Unrescue server: JSON request

```
{
  "unrescue": null
}
```

Response

There is no body content for the response of a successful POST operation.

POST

/servers/{server_id}/action

Add (Associate) Fixed Ip (Addfixedip Action)

Adds a fixed IP address to a server instance, which associates that address with the server. The fixed IP address is retrieved from the network that you specify in the request.

Specify the **addFixedIp** action and the network ID in the request body.

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

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|------------|------|--------|-------------------------|
| server_id | body | string | The UUID of the server. |
| addFixedIp | body | string | The action. |

Example Add (Associate) Fixed Ip (Addfixedip Action): JSON request

```
{
  "addFixedIp": {
    "networkId": 1
  }
}
```

Response

POST

/servers/{server_id}/action

Remove (Disassociate) Fixed Ip (Removefixedip Action)

Removes, or disassociates, a fixed IP address from a server.

Specify the `removeFixedIp` action in the request body.

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

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|---------------|------|--------|-------------------------|
| server_id | body | string | The UUID of the server. |
| removeFixedIp | body | string | The action. |

Example Remove (Disassociate) Fixed Ip (Removefixedip Action): JSON request

```
{
  "removeFixedIp": {
    "address": "10.0.0.4"
  }
}
```

Response

POST

/servers/{server_id}/action

Evacuate Server (evacuate Action)

Evacuates a server from a failed host to a new host.

- Specify the `evacuate` action in the request body.
- In the request body, if `onSharedStorage` is set, then do not set `adminPass`.
- The target host should not be the same as the instance host.

Normal response codes: 200

Error response codes: computeFault(400, 500), serviceUnavailable(503), badRequest(400), unauthorized(401), forbidden(403), badMethod(405), itemNotFound(404)

Request

| Name | In | Type | Description |
|----------------------|------|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| server_id | body | string | The UUID of the server. |
| evacuate | body | string | Specify the <code>evacuate</code> action in the request body. |
| host (Optional) | body | string | The name or ID of the host to which the server is evacuated. |
| adminPass (Optional) | body | string | An administrative password to access the evacuated or rebuilt instance. To set the administrative password, set the <code>enable_instance_password</code> configuration option to <code>True</code> . If you set this option to <code>False</code> and you try to set the administrative password, the API does not set the password and the response shows a <code>null</code> value for the <code>adminPass</code> parameter. |
| onSharedStorage | body | boolean | Server on shared storage. Deprecated in version 2.14 |
| force (Optional) | body | boolean | Force an evacuation by not verifying the provided destination host by the scheduler. New in version 2.29 |

Example Evacuate Server (evacuate Action):

```
{
  "evacuate": {
    "host": "b419863b7d814906a68fb31703c0dbd6",
    "adminPass": "MySecretPass",
    "onSharedStorage": "False"
  }
}
```

Response

| Name | In | Type | Description |
|----------------------|------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
| adminPass (Optional) | body | string | An administrative password to access the evacuated or rebuilt instance. To set the administrative password, set the <code>enable_instance_password</code> |

| Name | In | Type | Description |
|------|----|------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | configuration option to <code>True</code> . If you set this option to <code>False</code> and you try to set the administrative password, the API does not set the password and the response shows a <code>null</code> value for the <code>adminPass</code> parameter. Deprecated in version 2.14 |

Example Evacuate Server (evacuate Action):

```
{
  "adminPass": "MySecretPass"
}
```

POST

/servers/{server_id}/action

Force-Delete Server (forceDelete Action)

Force-deletes a server before deferred cleanup.

Specify the `forceDelete` action in the request body.

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

Normal response codes: 202

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. |
| forceDelete | body | string | The action. |

Example Force-Delete Server (forceDelete Action): JSON request

```
{
  "forceDelete": null
}
```

Response

No body is returned on a successful submission.

POST

/servers/{server_id}/action

Restore Soft-Deleted Instance (restore Action)

Restores a previously soft-deleted server instance. You cannot use this method to restore deleted instances.

Specify the `restore` action in the request body.

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

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|------------------------|------|--------|-------------------------|
| <code>server_id</code> | path | string | The UUID of the server. |
| <code>restore</code> | body | string | The action. |

Example Restore Soft-Deleted Instance (restore Action): JSON request

```
{
  "restore": null
}
```

Response

No body is returned on a successful submission.

POST

`/servers/{server_id}/action`

Show Console Output (Os-Getconsoleoutput Action)

Shows console output for a server instance.

This returns the text of the console since boot in a REST response. The console content may be large, you can limit it with the optional `length` parameter.

Normal response codes: 200

Error response codes: `unauthorized(401)`, `forbidden(403)`, `notFound(404)`, `conflict(409)`, `methodNotImplemented(501)`

Request

| Name | In | Type | Description |
|----------------------------------|------|--------|-------------------------|
| <code>server_id</code> | path | string | The UUID of the server. |
| <code>os-getConsoleOutput</code> | body | string | The action. |

| Name | In | Type | Description |
|-------------------|------|--------|----------------------------------------------------------------------------------------------------------------|
| length (Optional) | body | string | The number of lines to fetch from the end of console log. All lines will be returned if this is not specified. |

Example Get console output

This requests the last 50 lines of the server in question.

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

Response

| Name | In | Type | Description |
|--------|------|--------|---------------------------------------------------------------------------------------------------|
| output | body | string | The console output as a string. Control characters will be escaped to create a valid JSON string. |

Example Get console output: JSON response

```
{
  "output": "FAKE CONSOLE OUTPUT\nANOTHER\nLAST LINE"
}
```

POST

/servers/{server_id}/action

Get Rdp Console (os-getRDPConsole Action)

Gets an [RDP](#) console for a server.

Specify the `os-getRDPConsole` action in the request body.

Normal response codes: 200

Error response codes: `badRequest(400)`, `unauthorized(401)`, `forbidden(403)`, `itemNotFound(404)`, `conflict(409)`, `notImplemented(501)`

Request

| Name | In | Type | Description |
|------------------|------|--------|------------------------------------------------------------------------|
| server_id | body | string | The UUID of the server. |
| os-getRDPConsole | body | string | The action. |
| type | body | string | The type of RDP console. The correct value is <code>rdp-html5</code> . |

Example Get Rdp Console (os-getRDPConsole Action): JSON request

```
{
  "os-getRDPConsole": {
    "type": "rdp-html5"
  }
}
```

Response

Example Get Rdp Console (os-getRDPConsole Action): JSON response

```
{
  "console": {
    "type": "rdp-html5",
    "url": "http://127.0.0.1:6083/?token=191996c3-7b0f-42f3-95a7-f1839f2da6ed"
  }
}
```

POST

/servers/{server_id}/action

Get Serial Console (os-getSerialConsole Action)

Gets a serial console for a server.

Specify the `os-getSerialConsole` action in the request body.

Normal response codes: 200

Error response codes: `badRequest(400)`, `unauthorized(401)`, `forbidden(403)`, `itemNotFound(404)`, `conflict(409)`, `notImplemented(501)`

Request

| Name | In | Type | Description |
|---------------------|------|--------|------------------------------------------------------------------------|
| server_id | body | string | The UUID of the server. |
| os-getSerialConsole | body | string | The action. |
| type | body | string | The type of serial console. The correct value is <code>serial</code> . |

Example Get Serial Console (os-getSerialConsole Action): JSON request


```
{
  "os-getSerialConsole": {
    "type": "serial"
  }
}
```

Response

Example Get Serial Console (os-getSerialConsole Action): JSON response

```
{
  "console": {
    "type": "serial",
    "url": "ws://127.0.0.1:6083/?token=f9906a48-b71e-4f18-baca-c987da3ebdb3"
  }
}
```

POST

/servers/{server_id}/action

Get Spice Console (os-getSPICEConsole Action)

Gets a SPICE console for a server.

Specify the `os-getSPICEConsole` action in the request body.

Normal response codes: 200

Error response codes: `badRequest(400)`, `unauthorized(401)`, `forbidden(403)`, `itemNotFound(404)`, `conflict(409)`, `notImplemented(501)`

Request

| Name | In | Type | Description |
|--------------------|------|--------|----------------------------------------------------------------------------|
| server_id | body | string | The UUID of the server. |
| os-getSPICEConsole | body | string | The action. |
| type | body | string | The type of SPICE console. The correct value is <code>spice-html5</code> . |

Example Get Spice Console (os-getSPICEConsole Action): JSON request

```
{
  "os-getSPICEConsole": {
    "type": "spice-html5"
  }
}
```

Response

Example Get Spice Console (os-getSPICEConsole Action): JSON response

```
{
  "console": {
    "type": "spice-html5",
    "url": "http://127.0.0.1:6082/spice_auto.html?token=a30e5d08-6a20-4043-958f-"
  }
}
```

```
0852440c6af4"
  }
}
```

POST

/servers/{server_id}/action

Get Vnc Console (os-getVNCCConsole Action)

Gets a VNC console for a server.

Specify the `os-getVNCCConsole` action in the request body.

Normal response codes: 200

Error response codes: `badRequest(400)`, `unauthorized(401)`, `forbidden(403)`, `itemNotFound(404)`, `conflict(409)`, `notImplemented(501)`

Request

| Name | In | Type | Description |
|-------------------|------|--------|--------------------------------------------------------------------|
| server_id | body | string | The UUID of the server. |
| os-getVNCCConsole | body | string | The action. |
| type | body | string | The type of VNC console. The correct value is <code>novnc</code> . |

Example Get Vnc Console (os-getVNCCConsole Action): JSON request

```
{
  "os-getVNCCConsole": {
    "type": "novnc"
  }
}
```

Response

Example Get Vnc Console (os-getVNCCConsole Action): JSON response

```
{
  "console": {
    "type": "novnc",
    "url": "http://127.0.0.1:6080/vnc_auto.html?token=191996c3-7b0f-42f3-95a7-f1839f2da6ed"
  }
}
```

POST

/servers/{server_id}/action

Shelve Server (shelve Action)

Shelves a server.

Specify the `shelve` action in the request body.

All associated data and resources are kept but anything still in memory is not retained. To restore a shelved instance, use the `unshelve` action. To remove a shelved instance, use the `shelveOffload` action.

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

Preconditions

The server status must be `ACTIVE`, `SHUTOFF`, `PAUSED`, or `SUSPENDED`.

If the server is locked, you must have administrator privileges to shelve the server.

Asynchronous Postconditions

After you successfully shelve a server, its status changes to `SHELVED` and the image status is `ACTIVE`. The server instance data appears on the compute node that the Compute service manages.

If you boot the server from volumes or set the `shelved_offload_time` option to 0, the Compute service automatically deletes the instance on compute nodes and changes the server status to `SHELVED_OFFLOADED`.

Troubleshooting

If the server status does not change to `SHELVED` or `SHELVED_OFFLOADED`, the shelve operation failed. Ensure that you meet the preconditions and run the request again. If the request fails again, investigate whether another operation is running that causes a race condition.

Normal response codes: 202

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. |
| shelve | body | string | The action. |

Example Shelve server (shelve Action):

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

Response

No body is returned on successful request.

POST

`/servers/{server_id}/action`

Shelf-Offload (Remove) Server (shelveOffload Action)

Shelf-offloads, or removes, a shelved server.

Specify the `shelveOffload` action in the request body.

Data and resource associations are deleted. If an instance is no longer needed, you can remove that instance from the hypervisor to minimize resource usage.

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

Preconditions

The server status must be `SHELVED`.

If the server is locked, you must have administrator privileges to shelve-offload the server.

Asynchronous Postconditions

After you successfully shelve-offload a server, its status changes to `SHELVED_OFFLOADED`. The server instance data appears on the compute node.

Troubleshooting

If the server status does not change to `SHELVED_OFFLOADED`, the shelve-offload operation failed. Ensure that you meet the preconditions and run the request again. If the request fails again, investigate whether another operation is running that causes a race condition.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|----------------------------|-------------------|---------------------|-------------------------|
| <code>server_id</code> | <code>path</code> | <code>string</code> | The UUID of the server. |
| <code>shelveOffload</code> | <code>body</code> | <code>string</code> | The action. |

Example Shelf-Offload server (shelveOffload Action)

```
{
  "shelveOffload": null
}
```

Response

No body is returned on successful request.

POST

`/servers/{server_id}/action`

Unshelve (Restore) Shelved Server (unshelve Action)

Unshelves, or restores, a shelved server.

Specify the `unshelve` action in the request body.

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

Preconditions

The server status must be `SHELVED` or `SHELVED_OFFLOADED`.

If the server is locked, you must have administrator privileges to unshelve the server.

Asynchronous Postconditions

After you successfully shelve a server, its status changes to `ACTIVE`. The server appears on the compute node.

The shelved image is deleted from the list of images returned by an API call.

Troubleshooting

If the server status does not change to `ACTIVE`, the unshelve operation failed. Ensure that you meet the preconditions and run the request again. If the request fails again, investigate whether another operation is running that causes a race condition.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|------------------------|------|--------|-------------------------|
| <code>server_id</code> | path | string | The UUID of the server. |
| <code>unshelve</code> | body | string | The action. |

Example Unshelve server (unshelve Action):

```
{
  "unshelve": null
}
```

Response

No body is returned on successful request.

POST

`/servers/{server_id}/action`

Trigger Crash Dump In Server

New in version 2.17.

Trigger a crash dump in a server.

When a server starts behaving oddly at a fundamental level, it maybe be useful to get a kernel level crash dump to debug further. The crash dump action forces a crash dump followed by a system reboot of the server. Once the server comes back online, you can find a Kernel Crash Dump file in a certain location of the filesystem. For example, for Ubuntu you can find it in the `/var/crash` directory.

Warning

This action can cause data loss. Also, network connectivity can be lost both during and after this operation.

Normal response codes: 202

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

- 400 is returned if the server does not support a crash dump (either by configuration or because the backend does not support it)
- 409 is returned if the server is not in a state where a crash dump action is allowed.

Request

| Name | In | Type | Description |
|--------------------|------|--------|-------------------------------------------------------|
| server_id | path | string | The UUID of the server. |
| trigger_crash_dump | body | none | Specifies the trigger crash dump action should be run |

Example Trigger crash dump: JSON request

```
{
  "trigger_crash_dump": null
}
```

Response

No body is returned on a successful submission.

Servers - run an administrative action (servers, action)

Enables administrators to perform an action on a server. Specify the action in the request body.

You can change the administrative password for, create a back up of, and inject network information into a server.

You can migrate, live-migrate, reset networking on, and reset the state of a server.

POST

/servers/{server_id}/action

Create Server Back Up (createBackup Action)

Creates a back up of a server.

Specify the createBackup action in the request body.

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

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|---------------------|------|---------|-----------------------------------------------------------------------------------------------------------------|
| server_id | body | string | The UUID of the server. |
| createBackup | body | string | The action. |
| name | body | string | The name of the image to be backed up. |
| backup_type | body | string | The type of the backup, for example, daily. |
| rotation | body | integer | The rotation of the back up image, the oldest image will be removed when image count exceed the rotation count. |
| metadata (Optional) | body | object | Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each. |

Example Create Server Back Up (Createbackup Action): JSON request

```
{
  "createBackup": {
    "name": "Backup 1",
    "backup_type": "daily",
    "rotation": 1
  }
}
```

Response

On success a header 'Location' : <image location URL> and no body will be returned.

POST

/servers/{server_id}/action

Inject Network Information (injectNetworkInfo Action)

Injects network information into a server.

Specify the `injectNetworkInfo` action in the request body.

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

Warning

There is very limited support on this API, For more information, see [nova virt support matrix](#)

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|-------------------|------|--------|-------------------------|
| server_id | body | string | The UUID of the server. |
| injectNetworkInfo | body | string | The action. |

Example Inject Network Information (injectNetworkInfo Action): JSON request

```
{
  "injectNetworkInfo": null
}
```

Response

On success no body is returned.

POST

/servers/{server_id}/action

Migrate Server (migrate Action)

Migrates a server to a host. The scheduler chooses the host.

Specify the `migrate` action in the request body.

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: 202

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------|
| server_id | body | string | The UUID of the server. |
| migrate | body | string | The action. |

Example Migrate Server (migrate Action): JSON request

```
{
  "migrate": null
}
```

Response

On success no body is returned.

POST

/servers/{server_id}/action

Live-Migrate Server (os-migrateLive Action)

Live-migrates a server to a new host without rebooting.

Specify the `os-migrateLive` action in the request body.

Use the `host` parameter to specify the destination host. If you omit this parameter, the scheduler chooses a host. If a scheduled host is not suitable to do migration, the scheduler tries up to `migrate_max_retries` rescheduling attempts.

If both source and destination hosts provide local disks, you can set the `block_migration` parameter to `true`. If either host uses shared storage, the migration fails if you set this `block_migration` to `true`.

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

Starting from REST API version 2.34 pre-live-migration checks are done asynchronously, results of these checks are available in `instance-actions`. Nova responds immediately, and no pre-live-migration checks are returned. The instance will not immediately change state to `ERROR`, if a failure of the live-migration checks occurs.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|-----------|------|--------|----------------------------------------------------------------------------------------------------|
| server_id | body | string | The UUID of the server. |
| host | body | string | The host to which to migrate the server. If you omit this parameter, the scheduler chooses a host. |

| Name | In | Type | Description |
|------------------|------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| block_migration | body | boolean | Set to <code>True</code> to migrate local disks by using block migration. If the source or destination host uses shared storage and you set this value to <code>True</code> , the live migration fails. |
| disk_over_commit | body | boolean | Set to <code>True</code> to enable over commit when the destination host is checked for available disk space. Set to <code>False</code> to disable over commit. This setting affects only the libvirt virt driver. |
| force (Optional) | body | boolean | Force a live-migration by not verifying the provided destination host by the scheduler. New in version 2.30 |

Example Live-Migrate Server (os-migrateLive Action): JSON request

```
{
  "os-migrateLive": {
    "host": "01c0cadef72d47e28a672a76060d492c",
    "block_migration": false,
    "disk_over_commit": false
  }
}
```

Response

On success no body is returned.

POST

/servers/{server_id}/action

Reset Networking On A Server (resetNetwork Action)

Resets networking on a server.

Note

Only the XenServer driver implements this feature and only if the guest has the XenAPI agent in the targeted server.

Specify the `resetNetwork` action in the request body.

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: 202

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

Request

| Name | In | Type | Description |
|--------------|------|--------|-------------------------|
| server_id | body | string | The UUID of the server. |
| resetNetwork | body | string | The action. |

Example Reset Networking On A Server (resetNetwork Action): JSON request

```
{
  "resetNetwork": null
}
```

Response

On success no body is returned.

POST

/servers/{server_id}/action

Reset Server State (os-resetState Action)

Resets the state of a server.

Specify the `os-resetState` action and the `state` in the request body.

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: 202

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

Request

| Name | In | Type | Description |
|---------------------|------|--------|-----------------------------------------------------------------------------------------|
| server_id | body | string | The UUID of the server. |
| os-resetState | body | string | The action. |
| os-resetState.state | body | string | The state of the server to be set, <code>active</code> or <code>error</code> are valid. |

Example Reset Server State (os-resetState Action): JSON request

```
{
  "os-resetState": {
    "state": "active"
  }
}
```

Response

On success no body is returned.

Servers Security Groups (servers, os-security-groups)

Lists Security Groups for a server instance.

GET

/servers/{server_id}/os-security-groups

List Security Groups By Server

Lists security groups for a server.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------|
| server_id | body | string | The UUID of the server. |

Response

| Name | In | Type | Description |
|-------------------------------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 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. |
| description | body | string | Security group description. |
| id | body | string | The security group name or UUID. |
| name | body | string | The security group name. |
| rules | body | array | A security group rules object. |
| tenant_id | body | string | The UUID of the tenant in a multi-tenancy cloud. |

Example List security groups by server: JSON response

```
{
  "security_groups": [
    {
      "description": "default",
      "id": 1,
      "name": "default",
      "rules": [],
      "tenant_id": "6f70656e737461636b20342065766572"
    }
  ]
}
```

Servers diagnostics (servers, diagnostics)

Shows the usage data for a server.

GET

/servers/{server_id}/diagnostics

Show Server Diagnostics

Shows basic usage data for a server.

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

Normal response codes: 200

Error response codes: unauthorized(401), forbidden(403), notfound(404), conflict(409), notimplemented(501)

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------|
| server_id | body | string | The UUID of the server. |

Response

The response format for diagnostics is backend hypervisor specific, and not well defined. This should be considered a debug interface only, and not relied upon by programmatic tools.

Example Server diagnostics

Below is an example of diagnostics for a libvirt based server instance.

```
{
  "cpu0_time": 173000000000,
  "memory": 524288,
  "vda_errors": -1,
  "vda_read": 262144,
  "vda_read_req": 112,
  "vda_write": 5778432,
  "vda_write_req": 488,
  "vnet1_rx": 2070139,
  "vnet1_rx_drop": 0,
  "vnet1_rx_errors": 0,
  "vnet1_rx_packets": 26701,
  "vnet1_tx": 140208,
  "vnet1_tx_drop": 0,
  "vnet1_tx_errors": 0,
  "vnet1_tx_packets": 662
}
```

Servers IPs (servers, ips)

Lists the IP addresses for an instance and shows details for an IP address.

GET

/servers/{server_id}/ips

List Ips

Lists IP addresses that are assigned to an instance.

Policy defaults enable only users with the administrative role or the owner of the server 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), itemNotFound(404)

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------|
| server_id | body | string | The UUID of the server. |

Response

| Name | In | Type | Description |
|---------------|------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| addresses | body | object | The addresses for the server. Addresses information is hidden for any server in a state set in the <code>osapi_hide_server_address_states</code> configuration option. By default, servers in <code>building</code> state hide their addresses information. See <code>nova.conf</code> configuration options < http://docs.openstack.org/liberty/config-reference/content/list-of-compute-config-options.html >`_. |
| network_label | body | string | The network label, such as <code>public</code> or <code>private</code> . |
| addr | body | string | The IP address. |
| version | body | string | The version. |

Example List Ips: JSON response

```
{
  "addresses": {
    "private": [
      {
        "addr": "192.168.0.3",
        "version": 4
      }
    ]
  }
}
```

GET

/servers/{server_id}/ips/{network_label}

Show Ip Details

Shows IP addresses details for a network label of a server instance.

Policy defaults enable only users with the administrative role or the owner of the server 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), itemNotFound(404)

Request

| Name | In | Type | Description |
|---------------|------|--------|--------------------------------------------------------------------------|
| server_id | body | string | The UUID of the server. |
| network_label | path | string | The network label, such as <code>public</code> or <code>private</code> . |

Response

| Name | In | Type | Description |
|---------------|------|--------|--------------------------------------------------------------------------|
| network_label | body | string | The network label, such as <code>public</code> or <code>private</code> . |
| addr | body | string | The IP address. |
| version | body | string | The version. |

Example Show Ip Details: JSON response

```
{
  "private": [
    {
      "addr": "192.168.0.3",
      "version": 4
    }
  ]
}
```

Server metadata (servers, metadata)

Lists metadata, creates or replaces one or more metadata items, and updates one or more metadata items for a server.

Shows details for, creates or replaces, and updates a metadata item, by key, for a server.

GET

/servers/{server_id}/metadata

List All Metadata

Lists all metadata for a server.

Policy defaults enable only users with the administrative role or the owner of the server 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), itemNotFound(404)

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------|
| server_id | body | string | The UUID of the server. |

Response

| Name | In | Type | Description |
|----------|------|--------|---------------------------------------------------------------------------------------------------|
| metadata | body | object | Metadata key and value pairs. The maximum size for each metadata key and value pair is 255 bytes. |

Example List All Metadata:

```
{
  "metadata": {
    "foo": "Foo Value"
  }
}
```

POST

/servers/{server_id}/metadata

Update Metadata Items

Updates one or more metadata items for a server.

Replaces metadata items that match keys. Does not modify items that are not in the request.

Policy defaults enable only users with the administrative role or the owner of the server 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), itemNotFound(404), conflict(409)

Request

| Name | In | Type | Description |
|-----------|------|--------|---------------------------------------------------------------------------------------------------|
| server_id | body | string | The UUID of the server. |
| metadata | body | object | Metadata key and value pairs. The maximum size for each metadata key and value pair is 255 bytes. |

Example Update Metadata Items:

```
{
  "metadata": {
    "foo": "Foo Value"
  }
}
```

Response

| Name | In | Type | Description |
|----------|------|--------|---------------------------------------------------------------------------------------------------|
| metadata | body | object | Metadata key and value pairs. The maximum size for each metadata key and value pair is 255 bytes. |

Example Update Metadata Items:

```
{
  "metadata": {
    "foo": "Foo Value"
  }
}
```

PUT

/servers/{server_id}/metadata

Create Or Replace Metadata Items

Creates or replaces one or more metadata items for a server.

Creates any metadata items that do not already exist in the server. Removes and completely replaces any metadata items that already exist in the server with the metadata items in the request.

Policy defaults enable only users with the administrative role or the owner of the server 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)`, `itemNotFound(404)`, `conflict(409)`

Request

| Name | In | Type | Description |
|-----------|------|--------|---------------------------------------------------------------------------------------------------|
| server_id | body | string | The UUID of the server. |
| metadata | body | object | Metadata key and value pairs. The maximum size for each metadata key and value pair is 255 bytes. |

Example Create Or Replace Metadata Items:

```
{
  "metadata": {
    "foo": "Foo Value"
  }
}
```

Response

| Name | In | Type | Description |
|----------|------|--------|---------------------------------------------------------------------------------------------------|
| metadata | body | object | Metadata key and value pairs. The maximum size for each metadata key and value pair is 255 bytes. |

Example Create Or Replace Metadata Items:

```
{
  "metadata": {
    "foo": "Foo Value"
  }
}
```

GET

/servers/{server_id}/metadata/{key}

Show Metadata Item Details

Shows details for a metadata item, by key, for a server.

Policy defaults enable only users with the administrative role or the owner of the server 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), itemNotFound(404)

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------------------|
| server_id | body | string | The UUID of the server. |
| key | path | string | The metadata item key, as a string. |

| Name | In | Type | Description |
|------|----|------|-----------------------------------|
| | | | Maximum length is 255 characters. |

Response

| Name | In | Type | Description |
|------|------|--------|---------------------------------------------------------------------------------------------------|
| meta | body | object | Metadata key and value pairs. The maximum size for each metadata key and value pair is 255 bytes. |

Example Show Metadata Item Details:

```
{
  "meta": {
    "foo": "Foo Value"
  }
}
```

PUT

/servers/{server_id}/metadata/{key}

Create Or Update Metadata Item

Creates or replaces a metadata item, by key, for a server.

Creates a metadata item that does not already exist in the server. Removes and completely replaces a metadata item that already exists in the server with the metadata item in the request.

Policy defaults enable only users with the administrative role or the owner of the server 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)`, `itemNotFound(404)`, `conflict(409)`

Request

| Name | In | Type | Description |
|-----------|------|--------|-----------------------------------------------------------------------|
| server_id | body | string | The UUID of the server. |
| key | path | string | The metadata item key, as a string. Maximum length is 255 characters. |

Example Create Or Update Metadata Item:

```
{
  "meta": {
    "foo": "Bar Value"
  }
}
```

Response

| Name | In | Type | Description |
|------|------|--------|---------------------------------------------------------------------------------------------------|
| meta | body | object | Metadata key and value pairs. The maximum size for each metadata key and value pair is 255 bytes. |

Example Create Or Update Metadata Item:

```
{
  "meta": {
    "foo": "Foo Value"
  }
}
```

DELETE

/servers/{server_id}/metadata/{key}

Delete Metadata Item

Deletes a metadata item, by key, from a server.

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

Normal response codes: 204

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-----------------------------------------------------------------------|
| server_id | body | string | The UUID of the server. |
| key | path | string | The metadata item key, as a string. Maximum length is 255 characters. |

Response

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

Servers actions (servers, os-instance-actions)

List actions and action details for a server.

GET

/servers/{server_id}/os-instance-actions

List Actions For Server

Lists actions for a server.

Action information of deleted instances can be returned for requests later than microversion 2.21.

Policy defaults enable only users with the administrative role or the owner of the server 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), itemNotFound(404)

Request

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

Response

| Name | In | Type | Description |
|-----------------|------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| instanceActions | body | array | List of the actions for the given instance. |
| action | body | string | The name of the action. |
| instance_uuid | body | string | The UUID of the server. |
| message | body | string | The error message message about this action when error occurred. |
| project_id | body | string | The UUID of the project that this server belongs to. |
| request_id | body | string | The request id generated when execute the API of this action. |
| start_time | body | string | The date and time when the action was started. 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. |

Example List Actions For Server: JSON response

```
{
  "instanceActions": [
    {
      "action": "resize",
      "instance_uuid": "b48316c5-71e8-45e4-9884-6c78055b9b13",
      "message": "",
      "project_id": "842",
      "request_id": "req-25517360-b757-47d3-be45-0e8d2a01b36a",
```

```

        "start_time": "2012-12-05T01:00:00.000000",
        "user_id": "789"
    },
    {
        "action": "reboot",
        "instance_uuid": "b48316c5-71e8-45e4-9884-6c78055b9b13",
        "message": "",
        "project_id": "147",
        "request_id": "req-3293a3f1-b44c-4609-b8d2-d81b105636b8",
        "start_time": "2012-12-05T00:00:00.000000",
        "user_id": "789"
    }
]
}

```

GET

/servers/{server_id}/os-instance-actions/{request_id}

Show Server Action Details

Shows details for a server action.

Action details of deleted instances can be returned for requests later than microversion 2.21.

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), itemNotFound(404)

Request

| Name | In | Type | Description |
|------------|------|--------|-------------------------|
| server_id | path | string | The UUID of the server. |
| request_id | path | string | The ID of the request. |

Response

| Name | In | Type | Description |
|----------------|------|--------|------------------------------------------------------------------|
| instanceAction | body | object | The instance action object. |
| action | body | string | The name of the action. |
| instance_uuid | body | string | The UUID of the server. |
| message | body | string | The error message message about this action when error occurred. |
| project_id | body | string | The UUID of the project that this server belongs to. |
| request_id | body | string | The request id generated when execute the API of this action. |

| Name | In | Type | Description |
|--------------------|------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| start_time | body | string | <p>The date and time when the action was started. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| user_id | body | string | The user ID of the user who owns the server. |
| events | body | array | The events occurred in this action. |
| events.event | body | string | The name of the event. |
| events.start_time | body | string | <p>The date and time when the event was started. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| events.finish_time | body | string | <p>The date and time when the event was finished. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| events.result | body | string | The result of the event. |
| events.traceback | body | string | The traceback stack if error occurred in this event. |

Example Show Server Action Details: JSON response

```
{
  "instanceAction": {
    "action": "reboot",
    "events": [
      {
        "event": "schedule",
        "finish_time": "2012-12-05T01:02:00.000000",
        "result": "Success",
        "start_time": "2012-12-05T01:00:02.000000",
        "traceback": ""
      }
    ]
  }
}
```

```
        "event": "compute_create",
        "finish_time": "2012-12-05T01:04:00.000000",
        "result": "Success",
        "start_time": "2012-12-05T01:03:00.000000",
        "traceback": ""
    }
],
"instance_uuid": "b48316c5-71e8-45e4-9884-6c78055b9b13",
"message": "",
"project_id": "147",
"request_id": "req-3293a3f1-b44c-4609-b8d2-d81b105636b8",
"start_time": "2012-12-05T00:00:00.000000",
"user_id": "789"
}
```

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.

GET

/servers/{server_id}/os-interface

List Port Interfaces

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. |

| Name | In | Type | Description |
|----------|------|--------|------------------|
| 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"
    }
  ]
}
```

POST

/servers/{server_id}/os-interface

Create 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 <code>interfaceAttachment</code> action in the request body. |
| port_id (Optional) | body | string | The ID of the port for which you want to create an interface. The <code>net_id</code> and <code>port_id</code> parameters are mutually exclusive. If you do not specify the <code>port_id</code> 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 |

| Name | In | Type | Description |
|-----------------------------------|------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | a port interface. The <code>net_id</code> and <code>port_id</code> parameters are mutually exclusive. If you do not specify the <code>net_id</code> parameter, the OpenStack Networking API v2.0 uses the network information cache that is associated with the instance. |
| <code>fixed_ips</code> (Optional) | body | array | Fixed IP addresses. If you request a specific fixed IP address without a <code>net_id</code> , the request returns a Bad Request (400) response code. |
| <code>ip_address</code> | body | string | The IP address. It is required when <code>fixed_ips</code> is specified. |

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"
  }
}
```

Response

| Name | In | Type | Description |
|----------------------------------|------|--------|-------------------------------------|
| <code>interfaceAttachment</code> | body | object | The interface attachment. |
| <code>fixed_ips</code> | body | array | Fixed IP addresses with subnet IDs. |
| <code>ip_address</code> | body | string | The IP address. |
| <code>subnet_id</code> | body | string | The UUID of the subnet. |
| <code>mac_addr</code> | body | string | The MAC address. |
| <code>net_id</code> | body | string | The network ID. |

| Name | In | Type | Description |
|------------|------|--------|-----------------|
| 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"
  }
}
```

GET

/servers/{server_id}/os-interface/{port_id}

Show Port Interface Details

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. |

| Name | In | Type | Description |
|----------|------|--------|------------------|
| 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"
  }
}
```

DELETE

/servers/{server_id}/os-interface/{port_id}

Detach Interface

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. |

Response

No body is returned on successful request.

Servers password (servers, os-server-password)

Shows the encrypted administrative password. Also, clears the encrypted administrative password for a server, which removes it from the metadata server.

GET

/servers/{server_id}/os-server-password

Show Server Password

Shows the administrative password for a server.

This operation calls the metadata service to query metadata information and does not read password information from the server itself.

The password saved in the metadata service is typically encrypted using the public SSH key injected into this server, so the SSH private key is needed to read the password.

Policy defaults enable only users with the administrative role or the owner of the server 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), itemNotFound(404)

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------|
| server_id | body | string | The UUID of the server. |

Response

| Name | In | Type | Description |
|---------------------|------|--------|---------------------------------------------|
| password (Optional) | body | string | The password returned from metadata server. |

Example Show Server Password: JSON response

```
{
  "password":
    "x1oz03wLCBRWAa2yDjCCVx8vwNPypxnypmRYDa/zEr1Q+EzPe1S/Gz6nfmC52m0l0SCRuU0mG7kqqgejPof6M7b0ezS387zjq4LSvvwp2
    8zUknzy4YzfFGhnHAda13TxUJ26pfQCYrq8UTzmKF2Bq8ioSEtVVzM0A96pDh8W2i7B0z6MdoiVyiEv/I1K2LsuipfxSJR7Wdke4zNXJjH
    HP2RfYsVbZ/k9ANu+Nz4iIH8/7Cacud/pphH7EjrY6a4RZNRjQskrHKYed0YERpotyjYk1eDtRe72GrSiXteqCM4biaQ5w3ruS+AcX//PX
    k3uJ5kC7d67fPXaVz4WaQRYMg=="
}
```

DELETE

/servers/{server_id}/os-server-password

Clear Admin Password

Clears the encrypted administrative password for a server, which removes it from the database.

This action does not actually change the instance server password.

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

Normal response codes: 204

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------|
| server_id | body | string | The UUID of the server. |

Response

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

Servers virtual interfaces (servers, os-virtual-interfaces)

Lists virtual interfaces for a server instance.

GET

`/servers/{server_id}/os-virtual-interfaces`

List Virtual Interfaces

Lists the virtual interfaces for an instance.

Policy defaults enable only users with the administrative role or the owner of the server to perform this operation. Change these permissions through the `policy.json` file.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------|
| server_id | body | string | The UUID of the server. |

Response

| Name | In | Type | Description |
|--------------------|------|-------|-------------------------------------------|
| virtual_interfaces | body | array | A <code>virtual_interfaces</code> object. |

| Name | In | Type | Description |
|-------------|------|--------|------------------------------------|
| id | body | string | The UUID of the virtual interface. |
| mac_address | body | string | The MAC address. |

Example List Virtual Interfaces: JSON response

```
{
  "virtual_interfaces": [
    {
      "id": "cec8b9bb-5d22-4104-b3c8-4c35db3210a6",
      "mac_address": "fa:16:3e:3c:ce:6f"
    }
  ]
}
```

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 instance, shows details for a volume attachment, and detaches a volume.

GET

/servers/{server_id}/os-volume_attachments

List volume attachments for an instance

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. |

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 (Optional) | body | string | The UUID of the attachment. |
| serverId | body | string | The UUID of the server. |

| Name | In | Type | Description |
|----------|------|--------|----------------------------------|
| | | | |
| 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"
    }
  ]
}
```

POST

/servers/{server_id}/os-volume_attachments

Attach a volume to an instance

Attach a volume to an instance.

Normal response codes: 200

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

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, id, serverId, 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. |

Example Attach a volume to an instance: JSON request

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

Response

| Name | In | Type | Description |
|------------------|------|--------|-----------------------------------------------------------------------------------------------------------|
| volumeAttachment | body | object | A dictionary representation of a volume attachment containing the fields, device, id, serverId, volumeId. |
| device | body | string | Name of the device such as, /dev/vdb. |
| id (Optional) | 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"
  }
}
```

GET

/servers/{server_id}/os-volume_attachments/{attachment_id}

Show a detail of a volume attachment

Show a detail of a volume attachment.

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. |
| attachment_id | path | string | The UUID of the attachment. |

Response

| Name | In | Type | Description |
|------------------|------|--------|-----------------------------------------------------------------------------------------------------------|
| volumeAttachment | body | object | A dictionary representation of a volume attachment containing the fields, device, id, serverId, volumeId. |
| device | body | string | Name of the device such as, /dev/vdb. |
| id (Optional) | 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 Show a detail of a volume attachment: JSON response

```
{
  "volumeAttachment": {
    "device": "/dev/sdd",
    "id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
    "serverId": "2390fb4d-1693-45d7-b309-e29c4af16538",
    "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f803"
  }
}
```

PUT

/servers/{server_id}/os-volume_attachments/{attachment_id}

Update a volume attachment

Update a volume attachment.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|------------------|------|--------|-----------------------------------------------------------------------------------------------------------|
| server_id | path | string | The UUID of the server. |
| attachment_id | path | string | The UUID of the attachment. |
| volumeAttachment | body | object | A dictionary representation of a volume attachment containing the fields, device, id, serverId, volumeId. |

| Name | In | Type | Description |
|----------|------|--------|------------------------------------------------------------------|
| volumeId | body | string | The UUID of the volume to attach instead of the attached volume. |

Example Update a volume attachment: JSON request

```
{
  "volumeAttachment": {
    "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f805"
  }
}
```

Response

No body is returned on successful request.

DELETE

/servers/{server_id}/os-volume_attachments/{attachment_id}

Detach a volume from an instance

Detach a volume from an instance.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|---------------|------|--------|-----------------------------|
| server_id | path | string | The UUID of the server. |
| attachment_id | path | string | The UUID of the attachment. |

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.

GET

/flavors

List 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 server attribute. Default attribute is <code>created</code> . 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 server <code>sort_key</code> attribute. |
| sort_dir (Optional) | query | string | Sort direction. A valid value is <code>asc</code> (ascending) or <code>desc</code> (descending). Default is <code>desc</code> . 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 server <code>sort_key</code> attribute. |
| limit (Optional) | query | integer | Requests a page size of items. Returns a number of items up to a limit value. Use the <code>limit</code> parameter to make an initial limited request and use the ID of the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request. |
| marker (Optional) | query | string | The ID of the last-seen item. Use the <code>limit</code> parameter to make an initial limited request and use the ID of the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request. |
| minDisk (Optional) | query | integer | Filters the response by a minimum disk space, in GiB. For example, <code>100</code> . |
| minRam (Optional) | query | integer | Filters the response by a minimum RAM, in MB. For example, <code>512</code> . |
| is_public (Optional) | query | boolean | Filters the flavor list by only public flavors. By default <code>non-admin</code> users only see public flavors, and <code>admin</code> users can see additional non public flavors. |

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. |
| links | body | array | Links to the resources in question. See API Guide / Links and References for more info. |

Example List Flavors

Showing all the default flavors of a Liberty era Nova installation that was not customized by the site operators.

```
{
  "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"
    },
    {
      "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"
    },
    {
      "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"
    },
    {
      "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"
  },
  {
    "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"
  }
]
}

```

POST

/flavors

Create Flavor

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. |
| 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. |
| 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 it's limitations. Defaults to 0. |

| Name | In | Type | Description |
|---------------------------------------|------|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| swap (Optional) | body | integer | The size of a dedicated swap disk that will be allocated, in GiB. 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

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

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. |
| 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. |
| 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 References for more info. |

| Name | In | Type | Description |
|-------------------------------------|------|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 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 it's 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 GiB. If 0 (the default), no dedicated swap disk will be created. |
| 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. |

Example Create Flavor

```
{
  "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
  }
}
```

GET

/flavors/detail

List Flavors With Details

Lists flavors with details.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------------------------------|-------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>sort_key</code> (Optional) | query | string | Sorts by a server attribute. Default attribute is <code>created</code> . 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 server <code>sort_key</code> attribute. |
| <code>sort_dir</code> (Optional) | query | string | Sort direction. A valid value is <code>asc</code> (ascending) or <code>desc</code> (descending). Default is <code>desc</code> . 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 server <code>sort_key</code> attribute. |
| <code>limit</code> (Optional) | query | integer | Requests a page size of items. Returns a number of items up to a limit value. Use the <code>limit</code> parameter to make an initial limited request and use the ID of the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request. |
| <code>marker</code> (Optional) | query | string | The ID of the last-seen item. Use the <code>limit</code> parameter to make an initial limited request and use the ID of the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request. |
| <code>minDisk</code> (Optional) | query | integer | Filters the response by a minimum disk space, in GiB. For example, <code>100</code> . |
| <code>minRam</code> (Optional) | query | integer | Filters the response by a minimum RAM, in MB. For example, <code>512</code> . |
| <code>is_public</code> (Optional) | query | boolean | Filters the flavor list by only public flavors. By default <code>non_admin</code> users only see public flavors, and <code>admin</code> users can see additional non public flavors. |

Response

| Name | In | Type | Description |
|----------------------|------|--------|-----------------------------------------------------------------------------------------------|
| <code>flavors</code> | body | array | An array of flavor objects. |
| <code>name</code> | body | string | The display name of a flavor. |
| <code>id</code> | body | string | The ID of the flavor. While people often make this look like an int, this is really a string. |

| Name | In | Type | Description |
|-------------------------------------|------|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 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. |
| 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 References for 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 it's 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 GiB. If 0 (the default), no dedicated swap disk will be created. |
| 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. |

Example List Flavors With Details

```
{
  "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
    },
    {
      "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
},
{
  "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
},
{
  "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
},
{
  "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
}

```

}

GET

/flavors/{flavor_id}

Show Flavor Details

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. |
| 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. |
| 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 References for 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 it's 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. |

| Name | In | Type | Description |
|----------------------------|------|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| swap | body | integer | The size of a dedicated swap disk that will be allocated, in GiB. If 0 (the default), no dedicated swap disk will be created. |
| 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. |

Example Show Flavor Details

```
{
  "flavor": {
    "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
  }
}
```

DELETE

/flavors/{flavor_id}

Delete Flavor

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.

Flavors access (flavors, os-flavor-access)

Lists tenants who have access to a private flavor and adds private flavor access to and removes private flavor access from tenants. By default, only administrators can manage private flavor access. A private flavor has `is_public` set to `false` while a public flavor has `is_public` set to `true`.

GET

/flavors/{flavor_id}/os-flavor-access

List Flavor Access Information For Given Flavor

Lists flavor access information.

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_access | body | array | A list of objects, each with the keys <code>flavor_id</code> and <code>tenant_id</code> . |
| tenant_id | body | string | The UUID of the tenant in a multi-tenancy cloud. |
| flavor_id | body | string | The ID of the flavor. While people often make this look like an int, this is really a string. |

Example List Flavor Access Information For Given Flavor: JSON response

```
{
  "flavor_access": [
    {
      "flavor_id": "10",
      "tenant_id": "fake_tenant"
    }
  ]
}
```

POST

/flavors/{flavor_id}/action

Add Flavor Access To Tenant (addTenantAccess Action)

Adds flavor access to a tenant and flavor.

Specify the addTenantAccess action and the tenant in the request body.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------------|------|--------|--------------------------------------------------|
| flavor_id | path | string | The ID of the flavor. |
| addTenantAccess | body | string | The action. |
| tenant | body | string | The UUID of the tenant in a multi-tenancy cloud. |

Example Add Flavor Access To Tenant: JSON request

```
{
  "addTenantAccess": {
    "tenant": "fake_tenant"
  }
}
```

Response

| Name | In | Type | Description |
|---------------|------|--------|-----------------------------------------------------------------------------------------------|
| flavor_access | body | array | A list of objects, each with the keys flavor_id and tenant_id. |
| tenant_id | body | string | The UUID of the tenant in a multi-tenancy cloud. |
| flavor_id | body | string | The ID of the flavor. While people often make this look like an int, this is really a string. |

Example Add Flavor Access To Tenant: JSON response

```
{
  "flavor_access": [
    {
      "flavor_id": "10",
      "tenant_id": "fake_tenant"
    }
  ]
}
```

POST

/flavors/{flavor_id}/action

Remove Flavor Access From Tenant (removeTenantAccess Action)

Removes flavor access from a tenant and flavor.

Specify the `removeTenantAccess` action and the `tenant` in the request body.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------------------|------|--------|--------------------------------------------------|
| flavor_id | path | string | The ID of the flavor. |
| removeTenantAccess | body | string | The action. |
| tenant | body | string | The UUID of the tenant in a multi-tenancy cloud. |

Example Remove Flavor Access From Tenant: JSON request

```
{
  "removeTenantAccess": {
    "tenant": "fake_tenant"
  }
}
```

Response

| Name | In | Type | Description |
|---------------|------|--------|-----------------------------------------------------------------------------------------------|
| flavor_access | body | array | A list of objects, each with the keys <code>flavor_id</code> and <code>tenant_id</code> . |
| tenant_id | body | string | The UUID of the tenant in a multi-tenancy cloud. |
| flavor_id | body | string | The ID of the flavor. While people often make this look like an int, this is really a string. |

Example Remove Flavor Access From Tenant: JSON response

```
{
  "flavor_access": [
  ]
}
```


Flavors extra-specs (flavors, os-flavor-extra-specs)

Lists, creates, deletes, and updates the extra-specs or keys for a flavor.

GET

/flavors/{flavor_id}/os-extra_specs

List Extra Specs For A Flavor

Lists all extra specs for a flavor, by ID.

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 |
|-------------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| extra_specs | body | object | A dictionary of the flavor’s extra-specs key-and-value pairs. It appears in the os-extra-specs’ “create” REQUEST body, as well as the os-extra-specs’ “create” and “list” RESPONSE body. |
| key | body | string | The extra spec key of a flavor. It appears in the os-extra-specs’ “create” and “update” REQUEST body, as well as the os-extra-specs’ “create”, “list”, “show”, and “update” RESPONSE body. |
| value | body | string | The extra spec value of a flavor. It appears in the os-extra-specs’ “create” and “update” REQUEST body, as well as the os-extra-specs’ “create”, “list”, “show”, and “update” RESPONSE body. |

Example List Extra Specs For A Flavor: JSON response

```
{
  "extra_specs": {
    "key1": "value1",
    "key2": "value2"
  }
}
```

POST

/flavors/{flavor_id}/os-extra_specs

Create Extra Specs For A Flavor

Creates extra specs for a flavor, by ID.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-------------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| flavor_id | path | string | The ID of the flavor. |
| extra_specs | body | object | A dictionary of the flavor's extra-specs key-and-value pairs. It appears in the os-extra-specs' "create" REQUEST body, as well as the os-extra-specs' "create" and "list" RESPONSE body. |
| key | body | string | The extra spec key of a flavor. It appears in the os-extra-specs' "create" and "update" REQUEST body, as well as the os-extra-specs' "create", "list", "show", and "update" RESPONSE body. |
| value | body | string | The extra spec value of a flavor. It appears in the os-extra-specs' "create" and "update" REQUEST body, as well as the os-extra-specs' "create", "list", "show", and "update" RESPONSE body. |

Example Create Extra Specs For A Flavor: JSON request

```
{
  "extra_specs": {
    "key1": "value1",
    "key2": "value2"
  }
}
```

Response

| Name | In | Type | Description |
|-------------|------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| extra_specs | body | object | A dictionary of the flavor's extra-specs key-and-value pairs. It appears in the os-extra-specs' "create" REQUEST body, as well as the os-extra-specs' "create" and "list" RESPONSE body. |
| key | body | string | The extra spec key of a flavor. It appears in the os-extra-specs' "create" and "update" REQUEST body, as well as the os-extra-specs' "create", "list", "show", and "update" RESPONSE body. |

| Name | In | Type | Description |
|-------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| value | body | string | The extra spec value of a flavor. It appears in the os-extra-specs' "create" and "update" REQUEST body, as well as the os-extra-specs' "create", "list", "show", and "update" RESPONSE body. |

Example Create Extra Specs For A Flavor: JSON response

```
{
  "extra_specs": {
    "key1": "value1",
    "key2": "value2"
  }
}
```

GET

/flavors/{flavor_id}/os-extra_specs/{flavor_extra_spec_key}

Show An Extra Spec For A Flavor

Shows an extra spec, by key, for a flavor, by ID.

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. |
| flavor_extra_spec_key | path | string | The extra spec key for the flavor. |

Response

| Name | In | Type | Description |
|-------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| key | body | string | The extra spec key of a flavor. It appears in the os-extra-specs' "create" and "update" REQUEST body, as well as the os-extra-specs' "create", "list", "show", and "update" RESPONSE body. |
| value | body | string | The extra spec value of a flavor. It appears in the os-extra-specs' "create" and "update" REQUEST body, as well as the os-extra-specs' "create", "list", "show", and "update" RESPONSE body. |

Example Show An Extra Spec For A Flavor: JSON response

```
{
  "key1": "value1"
}
```

PUT

/flavors/{flavor_id}/os-extra_specs/{flavor_extra_spec_key}

Update An Extra Spec For A Flavor

Updates an extra spec, by key, for a flavor, by ID.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------------------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| flavor_id | path | string | The ID of the flavor. |
| flavor_extra_spec_key | path | string | The extra spec key for the flavor. |
| key | body | string | The extra spec key of a flavor. It appears in the os-extra-specs' "create" and "update" REQUEST body, as well as the os-extra-specs' "create", "list", "show", and "update" RESPONSE body. |
| value | body | string | The extra spec value of a flavor. It appears in the os-extra-specs' "create" and "update" REQUEST body, as well as the os-extra-specs' "create", "list", "show", and "update" RESPONSE body. |

Example Update An Extra Spec For A Flavor: JSON request

```
{
  "key1": "new_value1"
}
```

Response

| Name | In | Type | Description |
|-------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| key | body | string | The extra spec key of a flavor. It appears in the os-extra-specs' "create" and "update" REQUEST body, as well as the os-extra-specs' "create", "list", "show", and "update" RESPONSE body. |
| value | body | string | The extra spec value of a flavor. It appears in the os-extra-specs' "create" and "update" REQUEST body, as well as the os-extra-specs' "create", "list", "show", and "update" RESPONSE body. |

Example Update An Extra Spec For A Flavor: JSON response

```
{
  "key1": "new_value1"
}
```

}

DELETE

/flavors/{flavor_id}/os-extra_specs/{flavor_extra_spec_key}

Delete An Extra Spec For A Flavor

Deletes an extra spec, by key, for a flavor, by ID.

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. |
| flavor_extra_spec_key | path | string | The extra spec key for the flavor. |

Response

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

Keypairs (keypairs)

Generates, imports, and deletes SSH keys.

GET

/os-keypairs

List Keypairs

Lists keypairs that are associated with the account.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------------------|-------|---------|-----------------------------------------------------------------------------------------------------------|
| user_id (Optional) | query | string | This allows administrative users to list keypairs of specified user ID. New in version 2.10 |
| limit (Optional) | query | integer | Requests a page size of items. Returns a number of items up to a limit value. Use the |

| Name | In | Type | Description |
|-------------------|-------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | <p><code>limit</code> parameter to make an initial limited request and use the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request.</p> <p>New in version 2.35</p> |
| marker (Optional) | query | string | <p>The last-seen item. Use the <code>limit</code> parameter to make an initial limited request and use the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request.</p> <p>New in version 2.35</p> |

Response

| Name | In | Type | Description |
|-------------|------|--------|---------------------------------------------------------------------------------------------------------------------------|
| keypairs | body | array | Array of Keypair objects |
| name | body | string | A name for the keypair which will be used to reference it later. |
| public_key | body | string | The keypair public key. |
| fingerprint | body | string | The fingerprint for the keypair. |
| type | body | string | <p>The type of the keypair. Allowed values are <code>ssh</code> or <code>x509</code></p> <p>New in version 2.2</p> |
| user_id | body | string | <p>The <code>user_id</code> for a keypair.</p> <p>New in version 2.10</p> |

Example List Keypairs: JSON response

```
{
  "keypairs": [
    {
      "keypair": {
        "fingerprint": "7e:eb:ab:24:ba:d1:e1:88:ae:9a:fb:66:53:df:d3:bd",
        "name": "keypair-50ca852e-273f-4cdc-8949-45feba200837",
        "public_key": "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQCKF3MX590r1Bs3dH5CU71NmvpbrgZxSpyGjlnE8F1kirnc/Up221pjznxqeoTAwTw034k7Dz6aY
IrZGmQwe2Tke084yqv1j45Dkyoj95fw/sZacm0cZNuL69E0bEGHdprfGJQajrpz22NQoCD8TFB8wv+8om9NH9Le6s+wPe98WC77KLw8qgf
QsbIey+JawPwL4067ZdL5xrypuRjfiPWjgy/VH85IXg/Z/GONZ2nxHgSShMkwqSFECAC5L3PHB+0+/12M/iikdatFSVGjpuHvKL0s3oe7m
6H10f1uSJ85BzLWBbvva93qkGmLg4ZAc8rPh20+YIsBUHNLLMM/oQp Generated-by-Nova\n"
      }
    }
  ]
}
```

POST

/os-keypairs

Create Or Import Keypair

Generates or imports a keypair.

Normal response codes: 200, 201

Note

The success status code was changed from 200 to 201 in version 2.2

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

Request

| Name | In | Type | Description |
|-----------------------|-------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| keypair | body | object | Keypair object |
| name | body | string | A name for the keypair which will be used to reference it later. |
| public_key (Optional) | body | string | The public ssh key to import. If you omit this value, a keypair is generated for you. |
| type (Optional) | query | string | The type of the keypair. Allowed values are ssh or x509 New in version 2.2 |
| user_id (Optional) | body | string | The user_id for a keypair. This allows administrative users to upload keys for other users than themselves. New in version 2.10 |

Example Create Or Import Keypair: JSON request

```
{
  "keypair": {
    "name": "keypair-d20a3d59-9433-4b79-8726-20b431d89c78",
    "public_key": "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQgQDx8nkQv/zgGgB4rMYmIf+6A4l6Rr+o/6lHBQdw5aYd44bd8JttDCE/F/pNRr0lRE+PiqSP08nDPH
w0010JemH9gYgnnFlyY3/0cJ02RhIPyyxYpv9FhY+2YiUkpwF0cLImyrxEsYXpD/0d3ac30bNH6Sw9JD9UZHycpSxsIbECHw==
Generated-by-Nova"
  }
}
```

Response

| Name | In | Type | Description |
|---------|------|--------|--------------------------------------|
| keypair | body | object | Keypair object |
| name | body | string | A name for the keypair which will be |

| Name | In | Type | Description |
|------------------------|------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | used to reference it later. |
| public_key | body | string | The keypair public key. |
| private_key (Optional) | body | string | If you do not provide a public key on create, a new keypair will be built for you, and the private key will be returned during the initial create call. Make sure to save this, as there is no way to get this private key again in the future. |
| fingerprint | body | string | The fingerprint for the keypair. |
| type | body | string | The type of the keypair. Allowed values are Ssh or x509 New in version 2.2 |
| user_id | body | string | The user_id for a keypair. New in version 2.10 |

Example Create Or Import Keypair: JSON response

```
{
  "keypair": {
    "fingerprint": "1e:2c:9b:56:79:4b:45:77:f9:ca:7a:98:2c:b0:d5:3c",
    "name": "keypair-803a1926-af78-4b05-902a-1d6f7a8d9d3e",
    "public_key": "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQGDx8nkQv/zgGgB4rMYmIf+6A4l6Rr+o/6lHBQdw5aYd44bd8JttDCE/F/pNRR0lRE+PiQSP08nDPH
w0010JeMH9gYgmnFlyY3/0cJ02RhIPyyxYpv9FhY+2YiUkpWFOcLImyrxEsYXpD/0d3ac30bNH6Sw9JD9UZHYcpSxsIbECHw==
Generated-by-Nova",
    "user_id": "fake"
  }
}
```

GET

/os-keypairs/{keypair_name}

Show Keypair Details

Shows details for a keypair that is associated with the account.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------------------|------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| keypair_name | path | string | The keypair name. |
| user_id (Optional) | body | string | The user_id for a keypair. This allows administrative users to upload keys for other users than themselves. New in version 2.10 |

Response

| Name | In | Type | Description |
|-------------|------|--------|------------------------------------------------------------------------------------------|
| keypair | body | object | Keypair object |
| name | body | string | A name for the keypair which will be used to reference it later. |
| public_key | body | string | The keypair public key. |
| fingerprint | body | string | The fingerprint for the keypair. |
| type | body | string | The type of the keypair. Allowed values are ssh or x509 New in version 2.2 |
| user_id | body | string | The user_id for a keypair. New in version 2.10 |

Example Show Keypair Details: JSON response

```
{
  "keypair": {
    "fingerprint": "44:fe:29:6e:23:14:b9:53:5b:65:82:58:1c:fe:5a:c3",
    "name": "keypair-6638abdb-c4e8-407c-ba88-c8dd7cc3c4f1",
    "public_key": "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQCA1HTrHCbb9NawNLSV8N6tSa8i637+EC2dA+lsdHHfQ1T54t+N0nHhJPlKWDLhc579j87vp6RDFri
FJ/smsTnDnf64012z0kBaJpJPH2zXrBkZFK6q2rmxydURzX/z0yLSCP77SFJ0fdXWH2hMsAusflGyryHGx20n+mZK6mDrxVzGxEz228dwQ
5G7Az50oZDWyGH2pqPvKjkiFRw0jwUKf3BbkP0QvANAC0k26cv16mNFpFJfI1N30C51UsZQtKGR01ptJoWiJYKccqhkAKuo902tg/qup58
J5kflNm7I61sy1mJon6SGqNUSfoQagqtBH6vd/tU1jnlwZ03uUroAL Generated-by-Nova\n",
    "user_id": "fake",
    "deleted": false,
    "created_at": "2014-05-07T12:06:13.681238",
    "updated_at": null,
    "deleted_at": null,
    "id": 1
  }
}
```

DELETE

/os-keypairs/{keypair_name}

Delete Keypair

Deletes a keypair.

Normal response codes: 202, 204

Note

The normal return code is 204 in version 2.2 to match the fact that no body content is returned.

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

Request

| Name | In | Type | Description |
|--------------------|------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| keypair_name | body | string | A name for the keypair which will be used to reference it later. |
| user_id (Optional) | body | string | The user_id for a keypair. This allows administrative users to upload keys for other users than themselves. New in version 2.10 |

Response

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

Limits (limits)

Shows rate and absolute limits for the project.

GET
/limits

Show Rate And Absolute Limits

Shows rate and absolute limits for the project.

Normal response codes: 200

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

Response

| Name | In | Type | Description |
|----------------------------------|------|---------|------------------------------------------------------------------------|
| limits | body | object | Data structure that contains both absolute limits within a deployment. |
| absolute | body | object | Name/value pairs that set quota limits within a deployment. |
| maxImageMeta | body | integer | The number of allowed metadata items for each instance. |
| maxPersonality | body | integer | The number of allowed injected files for each tenant. |
| maxPersonalitySize | body | integer | The number of allowed bytes of content for each injected file. |
| maxSecurityGroupRules (Optional) | body | integer | The number of allowed rules for each security group. |

| Name | In | Type | Description |
|------------------------------|------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| maxSecurityGroups (Optional) | body | array | One or more security groups. Specify the name of the security group in the <code>name</code> attribute. If you omit this attribute, the API creates the server in the <code>default</code> security group. |
| maxServerGroupMembers | body | integer | The number of allowed members for each server group. |
| maxServerGroups | body | integer | The number of allowed server groups for each tenant. |
| maxServerMeta | body | integer | The number of allowed metadata items for each instance. |
| maxTotalCores | body | integer | The number of allowed instance cores for each tenant. |
| maxTotalFloatingIps | body | integer | The number of allowed floating IP addresses for each tenant. |
| maxTotalInstances | body | integer | The number of allowed instances for each tenant. |
| maxTotalKeypairs | body | integer | The number of allowed key pairs for each user. |
| maxTotalRAMSize | body | integer | The amount of allowed instance RAM, in MB, for each tenant. |
| rate | body | array | An empty list for backwards compatibility purposes. |

Example Show Rate And Absolute Limits: JSON response

```
{
  "limits": {
    "absolute": {
      "maxImageMeta": 128,
      "maxPersonality": 5,
      "maxPersonalitySize": 10240,
      "maxSecurityGroupRules": 20,
      "maxSecurityGroups": 10,
      "maxServerMeta": 128,
      "maxTotalCores": 20,
      "maxTotalFloatingIps": 10,
      "maxTotalInstances": 10,
      "maxTotalKeypairs": 100,
      "maxTotalRAMSize": 51200,
      "maxServerGroups": 10,
      "maxServerGroupMembers": 10,
      "totalCoresUsed": 0,
      "totalInstancesUsed": 0,
      "totalRAMUsed": 0,
      "totalSecurityGroupsUsed": 0,
      "totalFloatingIpsUsed": 0,
      "totalServerGroupsUsed": 0
    },
    "rate": []
  }
}
```

```
}  
}
```

Guest agents (os-agents)

Creates, lists, updates, and deletes guest agent builds. Use guest agents to access files on the disk, configure networking, or run other applications or scripts in the guest while the agent is running. This hypervisor-specific extension is currently only for the Xen driver. Use of guest agents is possible only if the underlying service provider uses the Xen driver.

GET

/os-agents

List Agent Builds

Lists agent builds.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------------------|-------|--------|--------------------------------------------|
| hypervisor (Optional) | query | string | Filters the response by a hypervisor type. |

Response

| Name | In | Type | Description |
|--------------|------|---------|---------------------------------------------------------------------|
| agents | body | array | A list of guest agent objects. |
| agent_id | body | integer | The agent ID. |
| architecture | body | string | The name of the cpu architecture. |
| hypervisor | body | string | The hypervisor type for the agent. Currently only xen is supported. |
| md5hash | body | string | The MD5 hash. |
| os | body | string | The name of the operating system. |
| url | body | string | The URL associated with the agent. |
| version | body | string | The version. |

Example List Agent Builds: JSON response

```
{  
  "agents": [  
    {
```

```
        "agent_id": 1,
        "architecture": "x86",
        "hypervisor": "xen",
        "md5hash": "add6bb58e139be103324d04d82d8f545",
        "os": "os",
        "url": "http://example.com/path/to/resource",
        "version": "8.0"
    }
}
```

POST

/os-agents

Create Agent Build

Creates an agent build.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------------|------|--------|---------------------------------------------------------------------|
| agent | body | object | The guest agent object. |
| hypervisor | body | string | The hypervisor type for the agent. Currently only xen is supported. |
| os | body | string | The name of the operating system. |
| architecture | body | string | The name of the cpu architecture. |
| version | body | string | The version. |
| md5hash | body | string | The MD5 hash. |
| url | body | string | The URL associated with the agent. |

Example Create Agent Build: JSON request

```
{
  "agent": {
    "hypervisor": "xen",
    "os": "os",
    "architecture": "x86",
    "version": "8.0",
    "md5hash": "add6bb58e139be103324d04d82d8f545",
    "url": "http://example.com/path/to/resource"
  }
}
```

Response

| Name | In | Type | Description |
|--------------|------|---------|---------------------------------------------------------------------|
| agent | body | object | The guest agent object. |
| agent_id | body | integer | The agent ID. |
| architecture | body | string | The name of the cpu architecture. |
| hypervisor | body | string | The hypervisor type for the agent. Currently only xen is supported. |
| md5hash | body | string | The MD5 hash. |
| os | body | string | The name of the operating system. |
| url | body | string | The URL associated with the agent. |
| version | body | string | The version. |

Example Create Agent Build: JSON response

```
{
  "agent": {
    "agent_id": 1,
    "architecture": "x86",
    "hypervisor": "xen",
    "md5hash": "add6bb58e139be103324d04d82d8f545",
    "os": "os",
    "url": "http://example.com/path/to/resource",
    "version": "8.0"
  }
}
```

PUT

/os-agents/{agent_build_id}

Update Agent Build

Updates an agent build.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|----------------|------|--------|------------------------------------|
| agent_build_id | path | string | The id of the agent build. |
| para | body | object | The parameter object. |
| url | body | string | The URL associated with the agent. |

| Name | In | Type | Description |
|---------|------|--------|---------------|
| md5hash | body | string | The MD5 hash. |
| version | body | string | The version. |

Example Update Agent Build: JSON request

```
{
  "para": {
    "url": "http://example.com/path/to/resource",
    "md5hash": "add6bb58e139be103324d04d82d8f545",
    "version": "7.0"
  }
}
```

Response

| Name | In | Type | Description |
|----------|------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| agent | body | object | The guest agent object. |
| agent_id | body | string | The agent ID. (This is a bug of API, this should be integer type which is consistent with the responses of agent create and list. This will be fixed in later microversion.) |
| md5hash | body | string | The MD5 hash. |
| url | body | string | The URL associated with the agent. |
| version | body | string | The version. |

Example Update Agent Build: JSON response

```
{
  "agent": {
    "agent_id": "1",
    "md5hash": "add6bb58e139be103324d04d82d8f545",
    "url": "http://example.com/path/to/resource",
    "version": "7.0"
  }
}
```

DELETE

/os-agents/{agent_build_id}

Delete Agent Build

Deletes an existing agent build.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|----------------|------|--------|----------------------------|
| agent_build_id | path | string | The id of the agent build. |

Response

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

Host aggregates (os-aggregates)

Creates and manages host aggregates. An aggregate assigns metadata to groups of compute nodes. Aggregates are only visible to the cloud provider.

GET

/os-aggregates

List Aggregates

Lists all aggregates. Includes the ID, name, and availability zone for each aggregate.

Normal response codes: 200

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

Response

| Name | In | Type | Description |
|-------------------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| aggregates | body | array | The list of existing aggregates. |
| availability_zone | body | string | The availability zone of the host aggregate. |
| created_at | body | string | <p>The date and time when the resource was created. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| deleted_at | body | string | <p>The date and time when the resource was deleted. If the resource has not been deleted yet, this field will be null. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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</p> |

| Name | In | Type | Description |
|------------|------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | value is -05:00. |
| deleted | body | boolean | A boolean indicate whether this aggregate is deleted or not, if it has not been deleted, false will appear. |
| hosts | body | array | A list of host ids in this aggregate. |
| id | body | string | The ID of the host aggregate. |
| metadata | body | object | Metadata key and value pairs associate with the aggregate. |
| name | body | string | The name of the host aggregate. |
| updated_at | body | string | <p>The date and time when the resource was updated, if the resource has not been updated, this field will show as null. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |

Example List Aggregates: JSON response

```
{
  "aggregates": [
    {
      "availability_zone": "nova",
      "created_at": "2013-08-18T12:17:56.856455",
      "deleted": false,
      "deleted_at": null,
      "hosts": [],
      "id": 1,
      "metadata": {
        "availability_zone": "nova"
      },
      "name": "name",
      "updated_at": null
    }
  ]
}
```

POST

/os-aggregates

Create Aggregate

Creates an aggregate in an availability zone.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|---------------------------------|------|--------|----------------------------------------------|
| aggregate | body | object | The host aggregate object. |
| name | body | string | The name of the host aggregate. |
| availability_zone (Optional) | body | string | The availability zone of the host aggregate. |

Example Create Aggregate: JSON request

```
{
  "aggregate":
  {
    "name": "name",
    "availability_zone": "nova"
  }
}
```

Response

| Name | In | Type | Description |
|-------------------|------|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| aggregate | body | object | The host aggregate object. |
| availability_zone | body | string | The availability zone of the host aggregate. |
| created_at | body | string | <p>The date and time when the resource was created. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| deleted_at | body | string | <p>The date and time when the resource was deleted. If the resource has not been deleted yet, this field will be null, The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| deleted | body | boolean | A boolean indicate whether this aggregate is deleted or not, if it has not been deleted, false will appear. |
| id | body | string | The ID of the host aggregate. |

| Name | In | Type | Description |
|------------|------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| name | body | string | The name of the host aggregate. |
| updated_at | body | string | <p>The date and time when the resource was updated, if the resource has not been updated, this field will show as <code>null</code>. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |

Example Create Aggregate: JSON response

```
{
  "aggregate": {
    "availability_zone": "nova",
    "created_at": "2013-08-18T12:17:55.751757",
    "deleted": false,
    "deleted_at": null,
    "id": 1,
    "name": "name",
    "updated_at": null
  }
}
```

GET

/os-aggregates/{aggregate_id}

Show Aggregate Details

Shows details for an aggregate. Details include hosts and metadata.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------------|------|--------|-------------------|
| aggregate_id | path | string | The aggregate ID. |

Response

| Name | In | Type | Description |
|-------------------|------|--------|------------------------------------------------------|
| aggregate | body | object | The host aggregate object. |
| availability_zone | body | string | The availability zone of the host aggregate. |
| created_at | body | string | The date and time when the resource was created. The |

| Name | In | Type | Description |
|------------|------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | <p>date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| deleted_at | body | string | <p>The date and time when the resource was deleted. If the resource has not been deleted yet, this field will be null. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| deleted | body | boolean | A boolean indicate whether this aggregate is deleted or not, if it has not been deleted, false will appear. |
| hosts | body | array | An array of host information. |
| id | body | string | The ID of the host aggregate. |
| metadata | body | object | Metadata key and value pairs associate with the aggregate. |
| name | body | string | The name of the host aggregate. |
| updated_at | body | string | <p>The date and time when the resource was updated, if the resource has not been updated, this field will show as null. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |

Example Show Aggregate Details: JSON response

```
{
  "aggregate": {
    "availability_zone": "nova",
    "created_at": "2013-08-18T12:17:56.380226",
    "deleted": false,
    "deleted_at": null,
    "hosts": [],
    "id": 1,
```

```

    "metadata": {
        "availability_zone": "nova"
    },
    "name": "name",
    "updated_at": null
}

```

PUT

/os-aggregates/{aggregate_id}

Update Aggregate

Updates either or both the name and availability zone for an aggregate.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|------------------------------|------|--------|----------------------------------------------|
| aggregate_id | path | string | The aggregate ID. |
| aggregate | body | object | The host aggregate object. |
| name (Optional) | body | string | The name of the host aggregate. |
| availability_zone (Optional) | body | string | The availability zone of the host aggregate. |

Example Update Aggregate: JSON request

```

{
  "aggregate": {
    "name": "newname",
    "availability_zone": "nova2"
  }
}

```

Response

| Name | In | Type | Description |
|-------------------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------|
| aggregate | body | object | The host aggregate object. |
| availability_zone | body | string | The availability zone of the host aggregate. |
| created_at | 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 |

| Name | In | Type | Description |
|------------|------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | For example, 2015-08-27T09:49:58-05:00. The $\pm hh:mm$ value, if included, is the time zone as an offset from UTC. In the previous example, the offset value is -05:00. |
| deleted_at | body | string | <p>The date and time when the resource was deleted. If the resource has not been deleted yet, this field will be null. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss$\pm hh:mm$</p> <p>For example, 2015-08-27T09:49:58-05:00. The $\pm hh:mm$ value, if included, is the time zone as an offset from UTC. In the previous example, the offset value is -05:00.</p> |
| deleted | body | boolean | A boolean indicate whether this aggregate is deleted or not, if it has not been deleted, false will appear. |
| hosts | body | array | An array of host information. |
| id | body | string | The ID of the host aggregate. |
| metadata | body | object | Metadata key and value pairs associate with the aggregate. |
| name | body | string | The name of the host aggregate. |
| updated_at | body | string | <p>The date and time when the resource was updated, if the resource has not been updated, this field will show as null. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss$\pm hh:mm$</p> <p>For example, 2015-08-27T09:49:58-05:00. The $\pm hh:mm$ value, if included, is the time zone as an offset from UTC. In the previous example, the offset value is -05:00.</p> |

Example Update Aggregate: JSON response

```
{
  "aggregate": {
    "availability_zone": "nova2",
    "created_at": "2013-08-18T12:17:56.259751",
    "deleted": false,
    "deleted_at": null,
    "hosts": [],
    "id": 1,
    "metadata": {
      "availability_zone": "nova2"
    },
    "name": "newname",
    "updated_at": "2013-08-18T12:17:56.286720"
```

```
}  
}
```

DELETE

/os-aggregates/{aggregate_id}

Delete Aggregate

Deletes an aggregate.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------------|------|--------|-------------------|
| aggregate_id | path | string | The aggregate ID. |

Response

POST

/os-aggregates/{aggregate_id}/action

Add Host

Adds a host to an aggregate.

Specify the add_host action and host name in the request body.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------------|------|--------|----------------------------------------------------|
| aggregate_id | path | string | The aggregate ID. |
| add_host | body | object | The add_host object used to add host to aggregate. |
| host | body | string | The name of the host. |

Example Add Host: JSON request

```
{  
  "add_host": {  
    "host": "21549b2f665945baaa7101926a00143c"  
  }  
}
```

Response

| Name | In | Type | Description |
|-------------------|------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| aggregate | body | object | The host aggregate object. |
| availability_zone | body | string | The availability zone of the host aggregate. |
| created_at | body | string | <p>The date and time when the resource was created. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| deleted_at | body | string | <p>The date and time when the resource was deleted. If the resource has not been deleted yet, this field will be null. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| deleted | body | boolean | A boolean indicate whether this aggregate is deleted or not, if it has not been deleted, false will appear. |
| hosts | body | array | An array of host information. |
| id | body | string | The ID of the host aggregate. |
| metadata | body | object | Metadata key and value pairs associate with the aggregate. |
| name | body | string | The name of the host aggregate. |
| updated_at | body | string | <p>The date and time when the resource was updated, if the resource has not been updated, this field will show as null. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |

Example Add Host: JSON response


```
{
  "aggregate": {
    "availability_zone": "nova",
    "created_at": "2013-08-18T12:17:56.297823",
    "deleted": false,
    "deleted_at": null,
    "hosts": [
      "21549b2f665945baaa7101926a00143c"
    ],
    "id": 1,
    "metadata": {
      "availability_zone": "nova"
    },
    "name": "name",
    "updated_at": null
  }
}
```

POST

/os-aggregates/{aggregate_id}/action

Remove Host

Removes a host from an aggregate.

Specify the `remove_host` action and host name in the request body.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------------|------|--------|---------------------------------------------------------|
| aggregate_id | path | string | The aggregate ID. |
| remove_host | body | object | The add_host object used to remove host from aggregate. |
| host | body | string | The name of the host. |

Example Remove Host: JSON request

```
{
  "remove_host": {
    "host": "bf1454b3d71145d49fca2101c56c728d"
  }
}
```

Response

| Name | In | Type | Description |
|-------------------|------|--------|----------------------------------------------|
| aggregate | body | object | The host aggregate object. |
| availability_zone | body | string | The availability zone of the host aggregate. |

| Name | In | Type | Description |
|------------|------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| created_at | body | string | <p>The date and time when the resource was created. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| deleted_at | body | string | <p>The date and time when the resource was deleted. If the resource has not been deleted yet, this field will be null. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| deleted | body | boolean | A boolean indicate whether this aggregate is deleted or not, if it has not been deleted, false will appear. |
| hosts | body | array | An array of host information. |
| id | body | string | The ID of the host aggregate. |
| metadata | body | object | Metadata key and value pairs associate with the aggregate. |
| name | body | string | The name of the host aggregate. |
| updated_at | body | string | <p>The date and time when the resource was updated, if the resource has not been updated, this field will show as null. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |

Example Remove Host: JSON response

```
{
  "aggregate": {
    "availability_zone": "nova",
    "created_at": "2013-08-18T12:17:56.990581",
    "deleted": false,
    "deleted_at": null,
    "hosts": [],
    "id": 1,
    "metadata": {
      "availability_zone": "nova"
    },
  },
}
```

```
    "name": "name",  
    "updated_at": null  
  }  
}
```

POST

/os-aggregates/{aggregate_id}/action

Create Or Update Aggregate Metadata

Creates or replaces metadata for an aggregate.

Specify the `set_metadata` action in the request body.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------------|------|--------|-------------------------------------------------------------------------------------------------|
| aggregate_id | path | string | The aggregate ID. |
| set_metadata | body | object | The <code>set_metadata</code> object used to set metadata for host aggregate. |
| metadata | body | object | Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each. |

Example Create Or Update Aggregate Metadata: JSON request

```
{  
  "set_metadata":  
    {  
      "metadata":  
        {  
          "key": "value"  
        }  
      }  
}
```

Response

| Name | In | Type | Description |
|-------------------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------|
| aggregate | body | object | The host aggregate object. |
| availability_zone | body | string | The availability zone of the host aggregate. |
| created_at | 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 |

| Name | In | Type | Description |
|------------|------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | For example, 2015-08-27T09:49:58-05:00. The $\pm hh:mm$ value, if included, is the time zone as an offset from UTC. In the previous example, the offset value is -05:00. |
| deleted_at | body | string | <p>The date and time when the resource was deleted. If the resource has not been deleted yet, this field will be null. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss$\pm hh:mm$</p> <p>For example, 2015-08-27T09:49:58-05:00. The $\pm hh:mm$ value, if included, is the time zone as an offset from UTC. In the previous example, the offset value is -05:00.</p> |
| deleted | body | boolean | A boolean indicate whether this aggregate is deleted or not, if it has not been deleted, false will appear. |
| hosts | body | array | An array of host information. |
| id | body | string | The ID of the host aggregate. |
| metadata | body | object | Metadata key and value pairs associate with the aggregate. |
| name | body | string | The name of the host aggregate. |
| updated_at | body | string | <p>The date and time when the resource was updated, if the resource has not been updated, this field will show as null. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss$\pm hh:mm$</p> <p>For example, 2015-08-27T09:49:58-05:00. The $\pm hh:mm$ value, if included, is the time zone as an offset from UTC. In the previous example, the offset value is -05:00.</p> |

Example Create Or Update Aggregate Metadata: JSON response

```
{
  "aggregate": {
    "availability_zone": "nova",
    "created_at": "2013-08-18T12:17:55.959571",
    "deleted": false,
    "deleted_at": null,
    "hosts": [],
    "id": 1,
    "metadata": {
      "availability_zone": "nova",
      "key": "value"
    },
    "name": "name",
    "updated_at": "2013-08-18T12:17:55.986540"
  }
}
```

Assisted volume snapshots (os-assisted-volume-snapshots)

Creates and deletes snapshots through an emulator/hypervisor. Only qcow2 file format is supported.

This API is only implemented by the libvirt compute driver.

An internal snapshot that lacks storage such as NFS or GlusterFS can use an emulator/hypervisor to add the snapshot feature. This is used to enable snapshot of volumes on backends such as NFS or GlusterFS by storing data as qcow2 files on these volumes.

This API is only ever called by Cinder, where it is used to create a snapshot for drivers that extend the remotefs Cinder driver.

POST

/os-assisted-volume-snapshots

Create Assisted Volume Snapshots

Creates an assisted volume snapshot.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|---------------------------|------|--------|---------------------------------------------------------------------------------------------------|
| snapshot | body | object | A partial representation of a snapshot that is used to create a snapshot. |
| volume_id | body | string | The source volume ID. |
| create_info | body | object | Information for snapshot creation. |
| create_info.snapshot_id | body | string | The UUID for a snapshot. |
| create_info.type | body | string | The snapshot type. A valid value is qcow2. |
| create_info.new_file | body | string | The name of the qcow2 file that Block Storage creates, which becomes the active image for the VM. |
| create_info.id (Optional) | body | string | Its an arbitrary string that gets passed back to the user. |

Example Create Assisted Volume Snapshots: JSON request

```
{
  "snapshot": {
    "volume_id": "521752a6-acf6-4b2d-bc7a-119f9148cd8c",
```

```

    "create_info": {
      "snapshot_id": "421752a6-acf6-4b2d-bc7a-119f9148cd8c",
      "type": "qcow2",
      "new_file": "new_file_name",
      "id": "421752a6-acf6-4b2d-bc7a-119f9148cd8c"
    }
  }
}

```

Response

| Name | In | Type | Description |
|----------|------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| snapshot | body | object | A partial representation of a snapshot that is used to create a snapshot. |
| id | body | string | Its the same arbitrary string which was sent in request body. NOTE: This string is passed back to user as it is and not being used in Nova internally. So use snapshot_id instead for further operation on this snapshot. |
| volumeId | body | string | The source volume ID. |

Example Create Assisted Volume Snapshots: JSON response

```

{
  "snapshot": {
    "id": "421752a6-acf6-4b2d-bc7a-119f9148cd8c",
    "volumeId": "521752a6-acf6-4b2d-bc7a-119f9148cd8c"
  }
}

```

DELETE

/os-assisted-volume-snapshots/{snapshot_id}

Delete Assisted Volume Snapshot

Deletes an assisted volume snapshot.

To make this request, add the `delete_info` query parameter to the URI, as follows:

DELETE /os-assisted-volume-snapshots/421752a6-acf6-4b2d-bc7a-119f9148cd8c?
delete_info='{ "volume_id": "521752a6-acf6-4b2d-bc7a-119f9148cd8c" }'

Normal response codes: 204

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

Request

| Name | In | Type | Description |
|-------------|-------|--------|---------------------------------------------------------------------|
| snapshot_id | path | string | The UUID of the snapshot. |
| delete_info | query | string | Information for snapshot deletion. Include the ID of the associated |

| Name | In | Type | Description |
|------|----|------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | <p>volume. For example:</p> <pre>DELETE /os-assisted-volume-snapshots? delete_info='{ "volume_id": "521752a6-acf6-4b2d-bc7a-119f9148cd8c"}'</pre> |

Response

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

Availability zones (os-availability-zone)

Lists and gets detailed availability zone information.

An availability zone is created or updated by setting the `availability_zone` parameter in the `create`, `update`, or `create` or `update` methods of the Host Aggregates API. See [Host Aggregates](#) for more details.

GET

/os-availability-zone

Get Availability Zone Information

Lists availability zone information.

Normal response codes: 200

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

Response

| Name | In | Type | Description |
|----------------------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| availabilityZoneInfo | body | array | The list of availability zone information. |
| hosts | body | object | An object containing a list of host information. The host information is comprised of host and service objects. The service object returns three parameters representing the states of the service: <code>active</code> , <code>available</code> , and <code>updated_at</code> . |
| zoneName | body | string | The availability zone name. |
| zoneState | body | object | The current state of the availability zone. |

| Name | In | Type | Description |
|-----------|------|---------|-----------------------------------------------------|
| available | body | boolean | Returns true if the availability zone is available. |

Example Get availability zone information

```
{
  "availabilityZoneInfo": [
    {
      "hosts": null,
      "zoneName": "nova",
      "zoneState": {
        "available": true
      }
    }
  ]
}
```

GET

/os-availability-zone/detail

Get Detailed Availability Zone Information

Gets detailed availability zone information. 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 |
|----------------------|------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| availabilityZoneInfo | body | array | The list of availability zone information. |
| hosts | body | object | An object containing a list of host information. The host information is comprised of host and service objects. The service object returns three parameters representing the states of the service: <code>active</code> , <code>available</code> , and <code>updated_at</code> . |
| zoneName | body | string | The availability zone name. |
| zoneState | body | object | The current state of the availability zone. |
| available | body | boolean | Returns true if the availability zone is available. |

Example Get detailed availability zone information

```
{
  "availabilityZoneInfo": [
    {
      "hosts": {
        "conductor": {
          "nova-conductor": {
            "active": true,
            "available": true,
            "updated_at": null
          }
        },
        "consoleauth": {
          "nova-consoleauth": {
            "active": true,
            "available": true,
            "updated_at": null
          }
        },
        "network": {
          "nova-network": {
            "active": true,
            "available": true,
            "updated_at": null
          }
        },
        "scheduler": {
          "nova-scheduler": {
            "active": true,
            "available": true,
            "updated_at": null
          }
        }
      },
      "zoneName": "internal",
      "zoneState": {
        "available": true
      }
    },
    {
      "hosts": {
        "compute": {
          "nova-compute": {
            "active": true,
            "available": true,
            "updated_at": null
          }
        }
      },
      "zoneName": "nova",
      "zoneState": {
        "available": true
      }
    }
  ]
}
```

Cells (os-cells, capacities)

Adds neighbor cells, lists neighbor cells, and shows the capabilities of the local cell.

GET

/os-cells

List Cells

Lists cells.

Normal response codes: 200

Error response codes: badRequest(400), unauthorized(401), forbidden(403), NotImplemented(501)

Request

| Name | In | Type | Description |
|-----------|------|--------|--------------------------------------------------|
| tenant_id | path | string | The UUID of the tenant in a multi-tenancy cloud. |

Response

Example List Cells: JSON response

```
{
  "cells": [
    {
      "name": "cell1",
      "rpc_host": null,
      "rpc_port": null,
      "type": "child",
      "username": "username1"
    },
    {
      "name": "cell3",
      "rpc_host": null,
      "rpc_port": null,
      "type": "child",
      "username": "username3"
    },
    {
      "name": "cell5",
      "rpc_host": null,
      "rpc_port": null,
      "type": "child",
      "username": "username5"
    },
    {
      "name": "cell2",
      "rpc_host": null,
      "rpc_port": null,
      "type": "parent",
      "username": "username2"
    },
    {
      "name": "cell4",
      "rpc_host": null,
      "rpc_port": null,
      "type": "parent",
      "username": "username4"
    }
  ]
}
```

POST

/os-cells

Create Cell

Create a new cell.

Normal response code: 200

Error response codes: badRequest(400), unauthorized(401), forbidden(403), NotImplemented(501)

GET

/os-cells/capacities

Capacities

Retrieve capacities.

Error response codes: badRequest(400), unauthorized(401), forbidden(403), NotImplemented(501)

GET

/os-cells/detail

List Cells With Details

Lists cells with details of capabilities.

Normal response codes: 200

Error response codes: badRequest(400), unauthorized(401), forbidden(403), NotImplemented(501)

Response

Example List Cells With Details: JSON response

GET

/os-cells/info

Info For This Cell

Retrieve info about the current cell.

Normal response code: 200

Error response codes: badRequest(400), unauthorized(401), forbidden(403), NotImplemented(501)

GET

/os-cells/{cell_id}

Show Cell Data

Shows data for a cell.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|---------|------|--------|-----------------------|
| cell_id | path | string | The UUID of the cell. |

Response

Example Show Cell Data: JSON response

```
{
  "cell": {
    "name": "cell13",
    "rpc_host": null,
    "rpc_port": null,
    "type": "child",
    "username": "username3"
  }
}
```

PUT

/os-cells/{cell_id}

Update a Cell

Update an existing cell.

Normal response code: 200

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

DELETE

/os-cells/{cell_id}

Delete a Cell

Remove a cell.

Normal response code: 200

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

GET

/os-cells/{cell_id}/capacities

Show Cell Capacities

Shows capacities for a cell.

Normal response codes: 200,501

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

Request

| Name | In | Type | Description |
|---------|------|--------|-----------------------|
| cell_id | path | string | The UUID of the cell. |

Response

Example Show Cell Capacities: JSON response

```
{
  "cell": {
    "capacities": {
      "disk_free": {
        "total_mb": 1052672,
        "units_by_mb": {
          "0": 0,
          "163840": 5,
          "20480": 46,
          "40960": 23,
          "81920": 11
        }
      },
      "ram_free": {
        "total_mb": 7680,
        "units_by_mb": {
          "16384": 0,
          "2048": 3,
          "4096": 1,
          "512": 13,
          "8192": 0
        }
      }
    }
  }
}
```

Cloudpipe (os-cloudpipe)

Manages virtual VPNs for projects.

GET

/os-cloudpipe

List Cloudpipes

Lists cloudpipes.

Normal response codes: 200

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

Response

Example List Cloudpipes: JSON response

```
{
  "cloudpipes": [
    {
```

```
        "created_at": "2012-11-27T17:18:01Z",
        "instance_id": "27deecdb-baa3-4a26-9c82-32994b815b01",
        "internal_ip": "192.168.0.3",
        "project_id": "fa1765bd-a352-49c7-a6b7-8ee108a3cb0c",
        "public_ip": "127.0.0.1",
        "public_port": 22,
        "state": "down"
    }
}
```

POST

/os-cloudpipe

Create Cloudpipe

Creates a cloudpipe.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------------------|------|--------|-------------------------------------------------------------------------------------|
| project_id (Optional) | body | string | The UUID of the project. If omitted, the project ID defaults to the calling tenant. |

Example Create Cloudpipe: JSON request

```
{
  "cloudpipe": {
    "project_id": "059f21e3-c20e-4efc-9e7a-eba2ab3c6f9a"
  }
}
```

Response

Example Create Cloudpipe: JSON response

```
{
  "instance_id": "1e9b8425-34af-488e-b969-4d46f4a6382e"
}
```

POST

/os-cloudpipe/configure-project

Update Cloudpipe

Updates the virtual private network (VPN) IP address and port for a cloudpipe instance.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|----------|------|--------|---------------------|
| vpn_ip | body | string | The VPN IP address. |
| vpn_port | body | string | The VPN port. |

Example Update Cloudpipe: JSON request

```
{
  "configure_project": {
    "vpn_ip": "192.168.1.1",
    "vpn_port": "2000"
  }
}
```

Response

Server consoles (servers, os-consoles, os-console-auth-token)

Manages server consoles.

GET

/servers/{server_id}/consoles

Lists Consoles

Lists all consoles for a server instance.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------|
| server_id | body | string | The UUID of the server. |

Response

| Name | In | Type | Description |
|--------------|------|--------|------------------------------|
| consoles | body | array | The list of console objects. |
| console | body | object | The console object. |
| console_type | body | string | The type of the console. |
| id | body | string | The UUID of the console. |

Example List Consoles

```
{
  "consoles": [
    {
      "console": {
        "console_type": "fake",
        "id": 1
      }
    }
  ]
}
```

POST

/servers/{server_id}/consoles

Create Console

Creates a console for a server instance.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------|
| server_id | body | string | The UUID of the server. |

Response

No body content is returned on a successful POST.

GET

/servers/{server_id}/consoles/{console_id}

Show Console Details

Shows console details for a server instance.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|------------|------|--------|--------------------------|
| server_id | body | string | The UUID of the server. |
| console_id | path | string | The UUID of the console. |

Response

| Name | In | Type | Description |
|-----------------|------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| console | body | object | The console object. |
| console_type | body | string | The type of the console. |
| host (Optional) | body | string | The name or ID of the host. |
| id | body | string | The UUID of the console. |
| instance_name | body | string | The name of the instance. |
| password | body | string | The password for the console. |
| port (Optional) | body | string | To provision the server instance with a NIC for an already existing port, specify the port-id in the <code>port</code> attribute in a <code>networks</code> object. The port status must be <code>DOWN</code> . required if you omit the <code>uuid</code> attribute. |

Example Show Console Details

```
{
  "console": {
    "console_type": "fake",
    "host": "fake",
    "id": 1,
    "instance_name": "instance-00000001",
    "password": "C4jBpJ6x",
    "port": 5999
  }
}
```

DELETE

/servers/{server_id}/consoles/{console_id}

Delete Console

Deletes a console for a server instance.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------|
| server_id | body | string | The UUID of the server. |

| Name | In | Type | Description |
|------|------|--------|--------------------------|
| id | path | string | The UUID of the console. |

Response

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

GET

/servers/{server_id}/os-console-auth-token

Show Console Authentication Token

Given the console authentication token for a server instance, shows the related connection information.

This method used to be available just for the `rdp-html5` console type prior microversion 2.31. Now it's available for all console types.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------|
| server_id | body | string | The UUID of the server. |

Response

| Name | In | Type | Description |
|---------------------------------|------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| console | body | object | The console object. |
| instance_uuid | path | string | The UUID of the instance. |
| host (Optional) | body | string | The name or ID of the host. |
| port (Optional) | body | string | To provision the server instance with a NIC for an already existing port, specify the port-id in the <code>port</code> attribute in a <code>networks</code> object. The port status must be DOWN . required if you omit the <code>uuid</code> attribute. |
| internal_access_path (Optional) | body | string | The id representing the internal access path. |

Example Show Console Authentication Token

```
{
  "console": {
    "instance_uuid": "b48316c5-71e8-45e4-9884-6c78055b9b13",
```

```
    "host": "localhost",
    "port": 5900,
    "internal_access_path": "51af38c3-555e-4884-a314-6c8cdde37444"
  }
}
```

Hosts (os-hosts)

Manages physical hosts. Some virt drivers do not support all host functions. For more information, see [nova virt support matrix](#)

Policy defaults enable only users with the administrative role to perform all os-hosts related operations. Cloud providers can change these permissions through the `policy.json` file.

GET

/os-hosts

List Hosts

Lists hosts.

Normal response codes: 200

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

Response

| Name | In | Type | Description |
|-----------|------|--------|-------------------------------------------------------|
| hosts | body | array | An array of host information. |
| zone | body | string | The available zone of the host. |
| host_name | body | string | The name of the host. |
| service | body | string | The name of the service which is running on the host. |

Example List Hosts

```
{
  "hosts": [
    {
      "host_name": "b6e4adbc193d428ea923899d07fb001e",
      "service": "conductor",
      "zone": "internal"
    },
    {
      "host_name": "09c025b0efc64211bd23fc50fa974cdf",
      "service": "compute",
      "zone": "nova"
    },
    {
      "host_name": "e73ec0bd35c64de4a1adfa8b8969a1f6",
      "service": "consoleauth",
      "zone": "internal"
    }
  ],
}
```

```
{
  "host_name": "396a8a0a234f476eb05fb9fbc5802ba7",
  "service": "network",
  "zone": "internal"
},
{
  "host_name": "abffda96592c4eacaf4111c28fddee17",
  "service": "scheduler",
  "zone": "internal"
}
]
```

GET

/os-hosts/{host_name}

Show Host Details

Shows details for a host.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-----------------------|
| host_name | path | string | The name of the host. |

Response

| Name | In | Type | Description |
|----------------|------|--------|------------------------------------------------------------------|
| host | body | array | The array that includes resource info of the host. |
| resource | body | object | The resource info of the host. |
| host.project | body | string | The project id (or special name like total, used_now, used_max). |
| host.cpu | body | string | The cpu info on the host. |
| host.memory_mb | body | string | The memory info on the host (in MB). |
| host.disk_gb | body | string | The disk info on the host (in GB). |
| host.host | body | string | The name of the host. |

Example Show Host Details

```
{
  "host": [
    {
      "resource": {
        "cpu": 1,
```

```

        "disk_gb": 1028,
        "host": "c1a7de0ac9d94e4baceae031d05caae3",
        "memory_mb": 8192,
        "project": "(total)"
    },
    {
        "resource": {
            "cpu": 0,
            "disk_gb": 0,
            "host": "c1a7de0ac9d94e4baceae031d05caae3",
            "memory_mb": 512,
            "project": "(used_now)"
        }
    },
    {
        "resource": {
            "cpu": 0,
            "disk_gb": 0,
            "host": "c1a7de0ac9d94e4baceae031d05caae3",
            "memory_mb": 0,
            "project": "(used_max)"
        }
    }
]
}

```

PUT

/os-hosts/{host_name}

Update Host status

Enables, disables a host or put a host in maintenance or normal mode.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------------------------|------|--------|-------------------------------------------------------------------------|
| host_name | path | string | The name of the host. |
| status (Optional) | path | string | The status of the current host, either 'enabled' or 'disabled'. |
| maintenance_mode (Optional) | path | string | Mode of maintenance state, either 'on_maintenance' or 'off_maintenance' |

Example Enable Host: JSON request

```

{
    "status": "enable",
    "maintenance_mode": "disable"
}

```

Response

| Name | In | Type | Description |
|------------------|------|--------|-------------------------------------------------------------------------|
| host | body | string | The name of the host. |
| status | body | string | The status of the current host, either 'enabled' or 'disabled'. |
| maintenance_mode | body | string | Mode of maintenance state, either 'on_maintenance' or 'off_maintenance' |

Example Enable Host

```
{
  "host": "65c5d5b7e3bd44308e67fc50f362aee6",
  "maintenance_mode": "off_maintenance",
  "status": "enabled"
}
```

GET

/os-hosts/{host_name}/reboot

Reboot Host

Reboots a host.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-----------------------|
| host_name | path | string | The name of the host. |

Response

| Name | In | Type | Description |
|--------------|------|--------|-------------------------------|
| host | body | string | The name of the host. |
| power_action | body | string | The power action on the host. |

Example Reboot Host: JSON response

```
{
  "host": "9557750dbc464741a89c907921c1cb31",
  "power_action": "reboot"
}
```

GET

/os-hosts/{host_name}/shutdown

Shut Down Host

Shuts down a host.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-----------------------|
| host_name | path | string | The name of the host. |

Response

| Name | In | Type | Description |
|--------------|------|--------|-------------------------------|
| host | body | string | The name of the host. |
| power_action | body | string | The power action on the host. |

Example Shut Down Host

```
{
  "host": "77cfa0002e4d45fe97f185968111b27b",
  "power_action": "shutdown"
}
```

GET

/os-hosts/{host_name}/startup

Start Host

Starts a host.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-----------------------|
| host_name | path | string | The name of the host. |

Response

| Name | In | Type | Description |
|------|------|--------|-----------------------|
| host | body | string | The name of the host. |

| Name | In | Type | Description |
|--------------|------|--------|-------------------------------|
| power_action | body | string | The power action on the host. |

Example Start Host

```
{
  "host": "4b392b27930343bbaa27fd5d8328a564",
  "power_action": "startup"
}
```

Hypervisors (os-hypervisors)

Lists all hypervisors, shows summary statistics for all hypervisors over all compute nodes, shows details for a hypervisor, and shows the uptime for a hypervisor.

GET

/os-hypervisors

List 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: 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 <code>limit</code> parameter to make an initial limited request and use the ID of the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request. New in version 2.33 |
| marker (Optional) | query | string | The ID of the last-seen item. Use the <code>limit</code> parameter to make an initial limited request and use the ID of the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request. New in version 2.33 |

Response

| Name | In | Type | Description |
|---------------------|------|---------|-----------------------------------------------------------|
| hypervisors | body | array | An array of hypervisor information. |
| hypervisor_hostname | body | string | The hypervisor host name. |
| id | body | integer | The id of the hypervisor. |
| 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. |

Example List Hypervisors: JSON response

```
{
  "hypervisors": [
    {
      "hypervisor_hostname": "fake-mini",
      "id": 1,
      "state": "up",
      "status": "enabled"
    }
  ]
}
```

GET

/os-hypervisors/detail

List Hypervisors Details

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: 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 <code>limit</code> parameter to make an initial limited request and use the ID of the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request. |

| Name | In | Type | Description |
|-------------------|-------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | New in version 2.33 |
| marker (Optional) | query | string | <p>The ID of the last-seen item. Use the <code>limit</code> parameter to make an initial limited request and use the ID of the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request.</p> <p>New in version 2.33</p> |

Response

| Name | In | Type | Description |
|----------------------|------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| hypervisors | body | array | An array of hypervisor information. |
| cpu_info | body | object | <p>A dictionary that contains cpu information like <code>arch</code>, <code>model</code>, <code>vendor</code>, <code>features</code> and <code>topology</code>. The content of this field is hypervisor specific.</p> <p>NOTE: Since version 2.28 <code>cpu_info</code> field is returned as a dictionary instead of string.</p> |
| current_workload | body | integer | The <code>current_workload</code> 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 <code>enabled</code> or <code>disabled</code> . |
| state | body | string | The state of the hypervisor. One of <code>up</code> or <code>down</code> . |
| disk_available_least | body | integer | The actual free disk on this hypervisor(in GB). |
| 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 GB). |

| Name | In | Type | Description |
|------------------------|------|---------|--------------------------------------------------------------------------------------------------------|
| free_ram_mb | body | integer | The free RAM in this hypervisor(in MB). |
| hypervisor_hostname | body | string | The hypervisor host name. |
| hypervisor_type | body | string | The hypervisor type. |
| hypervisor_version | body | integer | The hypervisor version. |
| id | body | integer | The id of the hypervisor. |
| local_gb | body | integer | The disk in this hypervisor(in GB). |
| local_gb_used | body | integer | The disk used in this hypervisor(in GB). |
| memory_mb | body | integer | The memory of this hypervisor(in MB). |
| memory_mb_used | body | integer | The memory used in this hypervisor(in MB). |
| running_vms | body | integer | The number of running vms on this hypervisor. |
| 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. |
| 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. |

Example List Hypervisors Details: JSON response

Note

This API sample is actual since microversion v2.28

```
{
  "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": "fake-mini",
    "hypervisor_type": "fake",
    "hypervisor_version": 1000,
    "id": 1,
    "local_gb": 1028,
    "local_gb_used": 0,
    "memory_mb": 8192,
    "memory_mb_used": 512,
    "running_vms": 0,
    "service": {
        "host": "e6a37ee802d74863ab8b91ade8f12a67",
        "id": 2,
        "disabled_reason": null
    },
    "vcpus": 1,
    "vcpus_used": 0
}
]
}

```

GET

/os-hypervisors/statistics

Show Hypervisor Statistics

Shows summary statistics for all 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 object of the hypervisors' statics. |
| 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 |

| Name | In | Type | Description |
|----------------------|------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | 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 this hypervisor(in GB). |
| free_disk_gb | body | integer | The free disk remaining on this hypervisor(in GB). |
| free_ram_mb | body | integer | The free RAM in this hypervisor(in MB). |
| local_gb | body | integer | The disk in this hypervisor(in GB). |
| local_gb_used | body | integer | The disk used in this hypervisor(in GB). |
| memory_mb | body | integer | The memory of this hypervisor(in MB). |
| memory_mb_used | body | integer | The memory used in this hypervisor(in MB). |
| running_vms | body | integer | The total number of running vms on all hypervisors. |
| vcpus | body | integer | The number of vcpu in this hypervisor. |
| vcpus_used | body | integer | The number of vcpu used in this hypervisor. |

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": 1,
    "vcpus_used": 0
  }
}
```

GET

/os-hypervisors/{hypervisor_id}

Show Hypervisor Details

Shows details for a hypervisor.

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), itemNotFound(404)

Request

| Name | In | Type | Description |
|---------------|------|--------|---------------------------|
| hypervisor_id | path | string | The ID of the hypervisor. |

Response

| Name | In | Type | Description |
|----------------------|------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| hypervisor | body | object | The hypervisor object. |
| cpu_info | body | object | A dictionary that contains cpu information like <code>arch</code> , <code>model</code> , <code>vendor</code> , <code>features</code> and <code>topology</code> . The content of this field is hypervisor specific. NOTE: Since version 2.28 <code>cpu_info</code> field is returned as a dictionary instead of string. |
| 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. |
| current_workload | body | integer | The <code>current_workload</code> 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 this hypervisor(in GB). |
| 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 GB). |
| free_ram_mb | body | integer | The free RAM in this hypervisor(in MB). |

| Name | In | Type | Description |
|------------------------|------|---------|--------------------------------------------------------------------------------------------------------|
| hypervisor_hostname | body | string | The hypervisor host name. |
| hypervisor_type | body | string | The hypervisor type. |
| hypervisor_version | body | integer | The hypervisor version. |
| id | body | integer | The id of the hypervisor. |
| local_gb | body | integer | The disk in this hypervisor(in GB). |
| local_gb_used | body | integer | The disk used in this hypervisor(in GB). |
| memory_mb | body | integer | The memory of this hypervisor(in MB). |
| memory_mb_used | body | integer | The memory used in this hypervisor(in MB). |
| running_vms | body | integer | The number of running vms on this hypervisor. |
| 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. |
| 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. |

Example Show Hypervisor Details: JSON response

Note

This API sample is actual since microversion v2.28

```
{
  "hypervisor": {
    "cpu_info": {
      "arch": "x86_64",
      "model": "Nehalem",
      "vendor": "Intel",
      "features": [
        "pge",
        "clflush"
      ],

```

```

        "topology": {
            "cores": 1,
            "threads": 1,
            "sockets": 4
        }
    },
    "state": "up",
    "status": "enabled",
    "current_workload": 0,
    "disk_available_least": 0,
    "host_ip": "1.1.1.1",
    "free_disk_gb": 1028,
    "free_ram_mb": 7680,
    "hypervisor_hostname": "fake-mini",
    "hypervisor_type": "fake",
    "hypervisor_version": 1000,
    "id": 1,
    "local_gb": 1028,
    "local_gb_used": 0,
    "memory_mb": 8192,
    "memory_mb_used": 512,
    "running_vms": 0,
    "service": {
        "host": "043b3cacf6f34c90a7245151fc8ebcda",
        "id": 2,
        "disabled_reason": null
    },
    "vcpus": 1,
    "vcpus_used": 0
}

```

GET

/os-hypervisors/{hypervisor_id}/uptime

Show Hypervisor Uptime

Shows the uptime for a hypervisor.

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), itemNotFound(404), NotImplemented(501)

Request

| Name | In | Type | Description |
|---------------|------|--------|---------------------------|
| hypervisor_id | path | string | The ID of the hypervisor. |

Response

| Name | In | Type | Description |
|---------------------|------|--------|---------------------------|
| hypervisor | body | object | The hypervisor object. |
| hypervisor_hostname | body | string | The hypervisor host name. |

| Name | In | Type | Description |
|--------|------|---------|------------------------------------------------------------------------|
| id | body | integer | The id of the hypervisor. |
| 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. |
| uptime | body | string | The total uptime of the hypervisor and information about average load. |

Example Show Hypervisor Uptime: JSON response

```
{
  "hypervisor": {
    "hypervisor_hostname": "fake-mini",
    "id": 1,
    "state": "up",
    "status": "enabled",
    "uptime": " 08:32:11 up 93 days, 18:25, 12 users,  load average: 0.20, 0.12, 0.14"
  }
}
```

GET

/os-hypervisors/{hypervisor_id}/search

Search Hypervisor

Search hypervisor by given hypervisor id.

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 code: 200

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

Request

| Name | In | Type | Description |
|---------------|------|--------|---------------------------|
| hypervisor_id | path | string | The ID of the hypervisor. |

Response

| Name | In | Type | Description |
|---------------------|------|---------|-------------------------------------|
| hypervisors | body | array | An array of hypervisor information. |
| hypervisor_hostname | body | string | The hypervisor host name. |
| id | body | integer | The id of the hypervisor. |

| Name | In | Type | Description |
|--------|------|--------|-----------------------------------------------------------|
| 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. |

Example Search Hypervisor: JSON response

```
{
  "hypervisors": [
    {
      "hypervisor_hostname": "fake-mini",
      "id": 1,
      "state": "up",
      "status": "enabled"
    }
  ]
}
```

GET

/os-hypervisors/{hypervisor_id}/servers

List Hypervisor Servers

List all servers belong to given hypervisor.

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 code: 200

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

Request

| Name | In | Type | Description |
|---------------|------|--------|---------------------------|
| hypervisor_id | path | string | The ID of the hypervisor. |

Response

| Name | In | Type | Description |
|---------------------|------|---------|-------------------------------------------------|
| hypervisors | body | array | An array of hypervisor information. |
| hypervisor_hostname | body | string | The hypervisor host name. |
| id | body | integer | The id of the hypervisor. |
| state | body | string | The state of the hypervisor. One of up or down. |

| Name | In | Type | Description |
|--------------|------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| status | body | string | The status of the hypervisor. One of enabled or disabled. |
| servers | body | array | A list of server objects. |
| servers.uuid | body | string | The UUID of the server instance to which the API dispatches the event. You must assign this instance to a host. Otherwise, this call does not dispatch the event to the instance. |
| servers.name | body | string | The server name. |

Example List Hypervisor Servers: JSON response

```
{
  "hypervisors": [
    {
      "hypervisor_hostname": "fake-mini",
      "id": 1,
      "state": "up",
      "status": "enabled",
      "servers": [
        {
          "name": "test_server1",
          "uuid": "aaaaaaaa-aaaa-aaaa-aaaa-aaaaaaaaaaaaa"
        },
        {
          "name": "test_server2",
          "uuid": "bbbbbbbb-bbbb-bbbb-bbbb-bbbbbbbbbbbb"
        }
      ]
    }
  ]
}
```

Server usage audit log (os-instance-usage-audit-log)

Administrator only. Audit server usage of a tenant.

GET

/os-instance_usage_audit_log

List Server Usage Audits For An Tenant

Lists usage audits for all servers on the given tenant.

Normal response codes: 200

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

Response

| Name | In | Type | Description |
|---------------------------|------|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| instance_usage_audit_logs | body | object | The object of instance usage audit log information. |
| hosts_not_run | body | array | A list of the hosts whose instance audit tasks have not run. |
| log | body | object | The object of instance usage audit logs. |
| errors | body | integer | The number of errors. |
| instances | body | integer | The number of instances. |
| message | body | string | The log message of the instance usage audit task. |
| state | body | string | The state of the instance usage audit task. DONE or RUNNING . |
| num_hosts | body | integer | The number of the hosts. |
| num_hosts_done | body | integer | The number of the hosts whose instance audit tasks have been done. |
| num_hosts_not_run | body | integer | The number of the hosts whose instance audit tasks have not run. |
| num_hosts_running | body | integer | The number of the hosts whose instance audit tasks are running. |
| overall_status | body | string | <p>The overall status of instance audit tasks.</p> <p>M of N hosts done. K errors.</p> <p>The M value is the number of hosts whose instance audit tasks have been done in the period. The N value is the number of all hosts. The K value is the number of hosts whose instance audit tasks cause errors. If instance audit tasks have been done at all hosts in the period, the overall status is as follows:</p> <p>ALL hosts done. K errors.</p> |
| period_beginning | body | string | The beginning time of the instance usage audit period. For example, 2016-05-01 00:00:00. |
| period_ending | body | string | The ending time of the instance usage audit period. For example, 2016-06-01 00:00:00. |
| total_errors | body | integer | The total number of instance audit task errors. |
| total_instances | body | integer | The total number of VM instances in the period. |

Example List Usage Audits For All Servers on tenant

```
{
  "instance_usage_audit_logs": {
    "hosts_not_run": [
      "samplehost3"
    ],
    "log": {
      "samplehost0": {
        "errors": 1,
        "instances": 1,
        "message": "Instance usage audit ran for host samplehost0, 1 instances in 0.01 seconds.",
        "state": "DONE"
      },
      "samplehost1": {
        "errors": 1,
        "instances": 2,
        "message": "Instance usage audit ran for host samplehost1, 2 instances in 0.01 seconds.",
        "state": "DONE"
      },
      "samplehost2": {
        "errors": 1,
        "instances": 3,
        "message": "Instance usage audit ran for host samplehost2, 3 instances in 0.01 seconds.",
        "state": "DONE"
      }
    },
    "num_hosts": 4,
    "num_hosts_done": 3,
    "num_hosts_not_run": 1,
    "num_hosts_running": 0,
    "overall_status": "3 of 4 hosts done. 3 errors.",
    "period_beginning": "2012-06-01 00:00:00",
    "period_ending": "2012-07-01 00:00:00",
    "total_errors": 3,
    "total_instances": 6
  }
}
```

GET

/os-instance_usage_audit_log/{before_timestamp}

List Usage Audits Before Specified Time

Lists usage audits that occurred before a specified time.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|------------------|------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| before_timestamp | path | string | Filters the response by the date and time before which to list usage audits. The date and time stamp format is as follows: CCYY-MM-DD hh:mm:ss.NNNNNN For example, 2015-08-27 09:49:58 or 2015-08-27 09:49:58.123456. |

Response

| Name | In | Type | Description |
|--------------------------|------|--------|--------------------------------------------------------------|
| instance_usage_audit_log | body | object | The object of instance usage audit log information. |
| hosts_not_run | body | array | A list of the hosts whose instance audit tasks have not run. |

| Name | In | Type | Description |
|-------------------|------|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| log | body | object | The object of instance usage audit logs. |
| errors | body | integer | The number of errors. |
| instances | body | integer | The number of instances. |
| message | body | string | The log message of the instance usage audit task. |
| state | body | string | The state of the instance usage audit task. DONE or RUNNING . |
| num_hosts | body | integer | The number of the hosts. |
| num_hosts_done | body | integer | The number of the hosts whose instance audit tasks have been done. |
| num_hosts_not_run | body | integer | The number of the hosts whose instance audit tasks have not run. |
| num_hosts_running | body | integer | The number of the hosts whose instance audit tasks are running. |
| overall_status | body | string | <p>The overall status of instance audit tasks.</p> <p>M of N hosts done. K errors.</p> <p>The M value is the number of hosts whose instance audit tasks have been done in the period. The N value is the number of all hosts. The K value is the number of hosts whose instance audit tasks cause errors. If instance audit tasks have been done at all hosts in the period, the overall status is as follows:</p> <p>ALL hosts done. K errors.</p> |
| period_beginning | body | string | The beginning time of the instance usage audit period. For example, 2016-05-01 00:00:00. |
| period_ending | body | string | The ending time of the instance usage audit period. For example, 2016-06-01 00:00:00. |
| total_errors | body | integer | The total number of instance audit task errors. |
| total_instances | body | integer | The total number of VM instances in the period. |

Example List Usage Audits Before Specified Time

```
{
  "instance_usage_audit_log": {
    "hosts_not_run": [
      "samplehost3"
    ],
    "log": {
      "samplehost0": {
        "errors": 1,

```

```

        "instances": 1,
        "message": "Instance usage audit ran for host samplehost0, 1 instances in 0.01 seconds.",
        "state": "DONE"
    },
    "samplehost1": {
        "errors": 1,
        "instances": 2,
        "message": "Instance usage audit ran for host samplehost1, 2 instances in 0.01 seconds.",
        "state": "DONE"
    },
    "samplehost2": {
        "errors": 1,
        "instances": 3,
        "message": "Instance usage audit ran for host samplehost2, 3 instances in 0.01 seconds.",
        "state": "DONE"
    }
},
"num_hosts": 4,
"num_hosts_done": 3,
"num_hosts_not_run": 1,
"num_hosts_running": 0,
"overall_status": "3 of 4 hosts done. 3 errors.",
"period_beginning": "2012-06-01 00:00:00",
"period_ending": "2012-07-01 00:00:00",
"total_errors": 3,
"total_instances": 6
}
}

```

Migrations (os-migrations) (frozen)

Shows data on migrations.

Warning

The old top-level resource */os-migrations* is frozen, it won't be extended anymore. Use */servers/{uuid}/migrations* instead.

GET

/os-migrations

List Migrations

Lists in-progress migrations.

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), itemNotFound(404)

Request

| Name | In | Type | Description |
|---------------------------|-------|--------|-------------------------------------------------------------|
| status (Optional) | query | string | The status of migration to filter. |
| host (Optional) | query | string | The source/destination compute node of migration to filter. |
| source_compute (Optional) | query | string | The source compute node of migration to filter. |

| Name | In | Type | Description |
|---------------------------|-------|--------|-------------------------------------------------------------------|
| migration_type (Optional) | query | string | The type of migration to filter. |
| hidden (Optional) | query | string | The 'hidden' setting of migration to filter. |
| instance_uuid (Optional) | query | string | The uuid of the instance that migration is operated on to filter. |

Response

Example List Migrations: JSON response

```
{
  "migrations": [
    {
      "created_at": "2012-10-29T13:42:02.000000",
      "dest_compute": "compute2",
      "dest_host": "1.2.3.4",
      "dest_node": "node2",
      "id": 1234,
      "instance_uuid": "instance_id_123",
      "new_instance_type_id": 2,
      "old_instance_type_id": 1,
      "source_compute": "compute1",
      "source_node": "node1",
      "status": "Done",
      "updated_at": "2012-10-29T13:42:02.000000"
    },
    {
      "created_at": "2013-10-22T13:42:02.000000",
      "dest_compute": "compute20",
      "dest_host": "5.6.7.8",
      "dest_node": "node20",
      "id": 5678,
      "instance_uuid": "instance_id_456",
      "new_instance_type_id": 6,
      "old_instance_type_id": 5,
      "source_compute": "compute10",
      "source_node": "node10",
      "status": "Done",
      "updated_at": "2013-10-22T13:42:02.000000"
    }
  ]
}
```

Quota sets (os-quota-sets)

Permits administrators, depending on policy settings, to view default quotas, view details for quotas, revert quotas to defaults, and update the quotas for a project or a project and user.

GET

/os-quota-sets/{tenant_id}

Show A Quota

Show the quota for a project or a project and a user.

To show a quota for a project and a user, specify the `user_id` query parameter.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------------------|-------|--------|--------------------------------------------------|
| tenant_id | path | string | The UUID of the tenant in a multi-tenancy cloud. |
| user_id (Optional) | query | string | ID of user to list the quotas for. |

Response

| Name | In | Type | Description |
|-----------------------------|------|---------|-----------------------------------------------------------------------------------------------------------------------------|
| quota_set | body | string | A <code>quota_set</code> object. |
| cores | body | integer | The number of allowed instance cores for each tenant. |
| fixed_ips | body | integer | The number of allowed fixed IP addresses for each tenant. Must be equal to or greater than the number of allowed instances. |
| floating_ips | body | integer | The number of allowed floating IP addresses for each tenant. |
| id | body | string | The UUID of the tenant/user the quotas listed for. |
| injected_file_content_bytes | body | integer | The number of allowed bytes of content for each injected file. |
| injected_file_path_bytes | body | integer | The number of allowed bytes for each injected file path. |
| injected_files | body | integer | The number of allowed injected files for each tenant. |
| instances | body | integer | The number of allowed instances for each tenant. |
| key_pairs | body | integer | The number of allowed key pairs for each user. |
| metadata_items | body | integer | The number of allowed metadata items |

| Name | In | Type | Description |
|----------------------|------|---------|-------------------------------------------------------------|
| | | | for each instance. |
| ram | body | integer | The amount of allowed instance RAM, in MB, for each tenant. |
| security_group_rules | body | integer | The number of allowed rules for each security group. |
| security_groups | body | integer | The number of allowed security groups for each tenant. |
| server_groups | body | integer | The number of allowed server groups for each tenant. |
| server_group_members | body | integer | The number of allowed members for each server group. |

Example Show A Quota: JSON response

```
{
  "quota_set": {
    "cores": 20,
    "fixed_ips": -1,
    "floating_ips": 10,
    "id": "fake_tenant",
    "injected_file_content_bytes": 10240,
    "injected_file_path_bytes": 255,
    "injected_files": 5,
    "instances": 10,
    "key_pairs": 100,
    "metadata_items": 128,
    "ram": 51200,
    "security_group_rules": 20,
    "security_groups": 10,
    "server_groups": 10,
    "server_group_members": 10
  }
}
```

PUT

/os-quota-sets/{tenant_id}

Update Quotas

Update the quotas for a project or a project and a user.

Users can force the update even if the quota has already been used and the reserved quota exceeds the new quota.

To force the update, specify the "force": True attribute in the request

body, the default value is false.

To update a quota for a project and a user, specify the user_id query

parameter.

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. |
| user_id (Optional) | query | string | ID of user to set the quotas for. |
| quota_set | body | string | A quota_set object. |
| force (Optional) | body | boolean | You can force the update even if the quota has already been used and the reserved quota exceeds the new quota. To force the update, specify the "force": "True". Default is False. |
| cores (Optional) | body | integer | The number of allowed instance cores for each tenant. |
| fixed_ips (Optional) | body | integer | The number of allowed fixed IP addresses for each tenant. Must be equal to or greater than the number of allowed instances. |
| floating_ips (Optional) | body | integer | The number of allowed floating IP addresses for each tenant. |
| injected_file_content_bytes (Optional) | body | integer | The number of allowed bytes of content for each injected file. |
| injected_file_path_bytes (Optional) | body | integer | The number of allowed bytes for each injected file path. |
| injected_files (Optional) | body | integer | The number of allowed injected files for each tenant. |
| instances (Optional) | body | integer | The number of allowed instances for each tenant. |
| key_pairs (Optional) | body | integer | The number of allowed key pairs for each user. |
| metadata_items (Optional) | body | integer | The number of allowed metadata items for each instance. |

| Name | In | Type | Description |
|---------------------------------|------|---------|-------------------------------------------------------------|
| ram (Optional) | body | integer | The amount of allowed instance RAM, in MB, for each tenant. |
| security_group_rules (Optional) | body | integer | The number of allowed rules for each security group. |
| security_groups (Optional) | body | integer | The number of allowed security groups for each tenant. |
| server_groups (Optional) | body | integer | The number of allowed server groups for each tenant. |
| server_group_members (Optional) | body | integer | The number of allowed members for each server group. |

Example Update Quotas: JSON request

```
{
  "quota_set": {
    "security_groups": 45
  }
}
```

Example Update Quotas with the optional ``force`` attribute: JSON request

```
{
  "quota_set": {
    "force": "True",
    "instances": 45
  }
}
```

Response

| Name | In | Type | Description |
|-----------------------------|------|---------|-----------------------------------------------------------------------------------------------------------------------------|
| quota_set | body | string | A quota_set object. |
| cores | body | integer | The number of allowed instance cores for each tenant. |
| fixed_ips | body | integer | The number of allowed fixed IP addresses for each tenant. Must be equal to or greater than the number of allowed instances. |
| floating_ips | body | integer | The number of allowed floating IP addresses for each tenant. |
| injected_file_content_bytes | body | integer | The number of allowed bytes of content for each injected file. |
| injected_file_path_bytes | body | integer | The number of allowed bytes for each |

| Name | In | Type | Description |
|----------------------|------|---------|-------------------------------------------------------------|
| | | | injected file path. |
| injected_files | body | integer | The number of allowed injected files for each tenant. |
| instances | body | integer | The number of allowed instances for each tenant. |
| key_pairs | body | integer | The number of allowed key pairs for each user. |
| metadata_items | body | integer | The number of allowed metadata items for each instance. |
| ram | body | integer | The amount of allowed instance RAM, in MB, for each tenant. |
| security_group_rules | body | integer | The number of allowed rules for each security group. |
| security_groups | body | integer | The number of allowed security groups for each tenant. |
| server_groups | body | integer | The number of allowed server groups for each tenant. |
| server_group_members | body | integer | The number of allowed members for each server group. |

Example Update Quotas: JSON response

```
{
  "quota_set": {
    "cores": 20,
    "fixed_ips": -1,
    "floating_ips": 10,
    "injected_file_content_bytes": 10240,
    "injected_file_path_bytes": 255,
    "injected_files": 5,
    "instances": 10,
    "key_pairs": 100,
    "metadata_items": 128,
    "ram": 51200,
    "security_group_rules": 20,
    "security_groups": 45,
    "server_groups": 10,
    "server_group_members": 10
  }
}
```

DELETE

/os-quota-sets/{tenant_id}

Revert Quotas To Defaults

Reverts the quotas to default values for a project or a project and a user.

To revert quotas for a project and a user, specify the `user_id` query parameter.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|--------------------|-------|--------|--------------------------------------------------|
| tenant_id | path | string | The UUID of the tenant in a multi-tenancy cloud. |
| user_id (Optional) | query | string | ID of user to delete quotas for. |

Response

GET

/os-quota-sets/{tenant_id}/defaults

List Default Quotas For Tenant

Lists the default quotas for a project.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------|------|--------|--------------------------------------------------|
| tenant_id | path | string | The UUID of the tenant in a multi-tenancy cloud. |

Response

| Name | In | Type | Description |
|--------------|------|---------|-----------------------------------------------------------------------------------------------------------------------------|
| quota_set | body | string | A <code>quota_set</code> object. |
| cores | body | integer | The number of allowed instance cores for each tenant. |
| fixed_ips | body | integer | The number of allowed fixed IP addresses for each tenant. Must be equal to or greater than the number of allowed instances. |
| floating_ips | body | integer | The number of allowed floating IP addresses for each tenant. |

| Name | In | Type | Description |
|-----------------------------|------|---------|----------------------------------------------------------------|
| id | body | string | The UUID of the tenant/user the quotas listed for. |
| injected_file_content_bytes | body | integer | The number of allowed bytes of content for each injected file. |
| injected_file_path_bytes | body | integer | The number of allowed bytes for each injected file path. |
| injected_files | body | integer | The number of allowed injected files for each tenant. |
| instances | body | integer | The number of allowed instances for each tenant. |
| key_pairs | body | integer | The number of allowed key pairs for each user. |
| metadata_items | body | integer | The number of allowed metadata items for each instance. |
| ram | body | integer | The amount of allowed instance RAM, in MB, for each tenant. |
| security_group_rules | body | integer | The number of allowed rules for each security group. |
| security_groups | body | integer | The number of allowed security groups for each tenant. |
| server_groups | body | integer | The number of allowed server groups for each tenant. |
| server_group_members | body | integer | The number of allowed members for each server group. |

Example List Default Quotas For Tenant: JSON response

```
{
  "quota_set": {
    "cores": 20,
    "fixed_ips": -1,
    "floating_ips": 10,
    "id": "fake_tenant",
    "injected_file_content_bytes": 10240,
    "injected_file_path_bytes": 255,
    "injected_files": 5,
    "instances": 10,
    "key_pairs": 100,
    "metadata_items": 128,
    "ram": 51200,
    "security_group_rules": 20,
    "security_groups": 10,
    "server_groups": 10,
```

```
    "server_group_members": 10
  }
}
```

GET

/os-quota-sets/{tenant_id}/detail

Show The Detail of Quota

Show the detail of quota for a project or a project and a user.

To show a quota for a project and a user, specify the `user_id` query parameter.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------------------|-------|--------|--------------------------------------------------|
| tenant_id | path | string | The UUID of the tenant in a multi-tenancy cloud. |
| user_id (Optional) | query | string | ID of user to list the quotas for. |

Response

| Name | In | Type | Description |
|-----------------------------|------|--------|---------------------------------------------------------------------------------------------------------------------------------------|
| quota_set | body | string | A quota_set object. |
| cores | body | object | The object of detailed cores quota, including in_use, limit and reserved number of cores. |
| fixed_ips | body | object | The object of detailed fixed ips quota, including in_use, limit and reserved number of fixed ips. |
| floating_ips | body | object | The object of detailed floating ips quota, including in_use, limit and reserved number of floating ips. |
| id | body | string | The UUID of the tenant/user the quotas listed for. |
| injected_file_content_bytes | body | object | The object of detailed injected file content bytes quota, including in_use, limit and reserved number of injected file content bytes. |
| injected_file_path_bytes | body | object | The object of detailed injected file path bytes quota, including in_use, limit and |

| Name | In | Type | Description |
|----------------------|------|--------|-------------------------------------------------------------------------------------------------------------------------|
| | | | reserved number of injected file path bytes. |
| injected_files | body | object | The object of detailed injected files quota, including in_use, limit and reserved number of injected files. |
| instances | body | object | The object of detailed instances quota, including in_use, limit and reserved number of instances. |
| key_pairs | body | object | The object of detailed key pairs quota, including in_use, limit and reserved number of key pairs. |
| metadata_items | body | object | The object of detailed key metadata items quota, including in_use, limit and reserved number of metadata items. |
| ram | body | object | The object of detailed key ram quota, including in_use, limit and reserved number of ram. |
| security_group_rules | body | object | The object of detailed security group rules quota, including in_use, limit and reserved number of security group rules. |
| security_groups | body | object | The object of detailed security groups, including in_use, limit and reserved number of security groups. |
| server_groups | body | object | The object of detailed server groups, including in_use, limit and reserved number of server groups. |
| server_group_members | body | object | The object of detailed server group members, including in_use, limit and reserved number of server group members. |

Example Show A Quota: JSON response

```
{
  "quota_set": {
    "cores": {
      "in_use": 0,
      "limit": 20,
      "reserved": 0
    },
    "fixed_ips": {
      "in_use": 0,
      "limit": -1,
      "reserved": 0
    },
    "floating_ips": {
      "in_use": 0,
      "limit": 10,
      "reserved": 0
    },
    "id": "fake_tenant",
    "injected_file_content_bytes": {
      "in_use": 0,
      "limit": 10240,

```

```

        "reserved": 0
    },
    "injected_file_path_bytes": {
        "in_use": 0,
        "limit": 255,
        "reserved": 0
    },
    "injected_files": {
        "in_use": 0,
        "limit": 5,
        "reserved": 0
    },
    "instances": {
        "in_use": 0,
        "limit": 10,
        "reserved": 0
    },
    "key_pairs": {
        "in_use": 0,
        "limit": 100,
        "reserved": 0
    },
    "metadata_items": {
        "in_use": 0,
        "limit": 128,
        "reserved": 0
    },
    "ram": {
        "in_use": 0,
        "limit": 51200,
        "reserved": 0
    },
    "security_group_rules": {
        "in_use": 0,
        "limit": 20,
        "reserved": 0
    },
    "security_groups": {
        "in_use": 0,
        "limit": 10,
        "reserved": 0
    },
    "server_group_members": {
        "in_use": 0,
        "limit": 10,
        "reserved": 0
    },
    "server_groups": {
        "in_use": 0,
        "limit": 10,
        "reserved": 0
    }
}
}
}

```

Server groups (os-server-groups)

Lists, shows information for, creates, and deletes server groups.

GET

/os-server-groups

List Server Groups

Lists all server groups for the tenant.

Administrative users can use the `all_projects` query parameter to list all server groups for all projects.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|----------------------------|-------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| all_projects (Optional) | query | boolean | Administrator only. Lists server groups for all projects. For example: <code>GET /os-server-groups? all_projects=True</code> If you specify a tenant ID for a non-administrative user with this query parameter, the call lists all server groups for the tenant, or project, rather than for all projects. |

Response

| Name | In | Type | Description |
|--------------------------|-------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| server_groups | body | list | The list of existing server groups. |
| id | path | string | The UUID of the server group. |
| name | body | string | The name of the server group. |
| policies | body | array | A list of one or more policy names to associate with the server group. The list must contain at least one policy name. The current valid policy names are <code>anti-affinity</code> , <code>affinity</code> , <code>soft-anti-affinity</code> and <code>soft-affinity</code> , the latter two were added in microversion 2.15. Each policy name must be a non-empty string with no leading or trailing spaces. Maximum length is 255 characters. |
| members | body | array | A list of members in the server group. |
| metadata | body | object | Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each. The associated metadata key and value pairs. |
| project_id (Optional) | body | string | The project ID who owns the server group. New in version 2.13 |
| user_id (Optional) | query | string | The user ID who owns the server group. New in version 2.13 |

Example List Server Groups: JSON response

```
{  
  "server_groups": [  
    {  
      "id": "616fb98f-46ca-475e-917e-2563e5a8cd19",  
      "name": "test",  
      "policies": ["anti-affinity"],  
    },  
  ],  
}
```

```

        "members": [],
        "metadata": {}
    }
]
}

```

POST

/os-server-groups

Create Server Group

Creates a server group.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|----------|------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| name | body | string | The security group name. |
| policies | body | array | A list of one or more policy names to associate with the server group. The list must contain at least one policy name. The current valid policy names are <code>anti-affinity</code> , <code>affinity</code> , <code>soft-anti-affinity</code> and <code>soft-affinity</code> , the latter two were added in microversion 2.15. Each policy name must be a non-empty string with no leading or trailing spaces. Maximum length is 255 characters. |

Example Create Server Group: JSON request

```

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

```

Response

| Name | In | Type | Description |
|---------------|------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| server_groups | body | list | The list of existing server groups. |
| id | path | string | The UUID of the server group. |
| name | body | string | The name of the server group. |
| policies | body | array | A list of one or more policy names to associate with the server group. The list must contain at least one policy name. The current valid policy names are <code>anti-affinity</code> , <code>affinity</code> , <code>soft-anti-affinity</code> and <code>soft-affinity</code> , the latter two were added in microversion 2.15. Each policy name must be a non-empty string with no |

| Name | In | Type | Description |
|--------------------------|-------|--------|----------------------------------------------------------------------------------------------------------------------------------------------|
| | | | leading or trailing spaces. Maximum length is 255 characters. |
| members | body | array | A list of members in the server group. |
| metadata | body | object | Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each. The associated metadata key and value pairs. |
| project_id (Optional) | body | string | The project ID who owns the server group. New in version 2.13 |
| user_id (Optional) | query | string | The user ID who owns the server group. New in version 2.13 |

Example Create Server Group: JSON response

```
{
  "server_group": {
    "id": "5bbcc3c4-1da2-4437-a48a-66f15b1b13f9",
    "name": "test",
    "policies": ["anti-affinity"],
    "members": [],
    "metadata": {}
  }
}
```

GET

/os-server-groups/{server_group_id}

Show Server Group Details

Shows details for a server group.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------------|------|--------|-------------------------------|
| server_group_id | path | string | The UUID of the server group. |

Response

| Name | In | Type | Description |
|---------------|------|--------|-------------------------------------|
| server_groups | body | list | The list of existing server groups. |
| id | path | string | The UUID of the server group. |

| Name | In | Type | Description |
|--------------------------|-------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | |
| name | body | string | The name of the server group. |
| policies | body | array | A list of one or more policy names to associate with the server group. The list must contain at least one policy name. The current valid policy names are anti-affinity , affinity , soft-anti-affinity and soft-affinity , the latter two were added in microversion 2.15. Each policy name must be a non-empty string with no leading or trailing spaces. Maximum length is 255 characters. |
| members | body | array | A list of members in the server group. |
| metadata | body | object | Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each. The associated metadata key and value pairs. |
| project_id (Optional) | body | string | The project ID who owns the server group. New in version 2.13 |
| user_id (Optional) | query | string | The user ID who owns the server group. New in version 2.13 |

Example Show Server Group Details: JSON response

```
{
  "server_group": {
    "id": "5bbcc3c4-1da2-4437-a48a-66f15b1b13f9",
    "name": "test",
    "policies": ["anti-affinity"],
    "members": [],
    "metadata": {}
  }
}
```

DELETE

/os-server-groups/{server_group_id}

Delete Server Group

Deletes a server group.

Normal response codes: 204

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

Request

| Name | In | Type | Description |
|-----------------|------|--------|-------------------------------|
| server_group_id | path | string | The UUID of the server group. |

Response

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

Server tags (servers, tags)

Lists tags, creates, replaces or deletes one or more tags for a server, checks the existence of a tag for a server.

Available since version 2.26

Tags have the following restrictions:

- Tag is a Unicode bytestring no longer than 60 characters.
- Tags are case sensitive.
- ‘/’ is not allowed to be in a tag name
- Comma is not allowed to be in a tag name in order to simplify requests that specify lists of tags
- All other characters are allowed to be in a tag name
- Each server can have up to 50 tags.

GET

/servers/{server_id}/tags

List Tags

Lists all tags for 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. |

Response

| Name | In | Type | Description |
|------|------|-------|--------------------------------------------------------------------------------------------------------|
| tags | body | array | <p>A list of tags. The maximum count of tags in this list is 50.</p> <p>New in version 2.26</p> |

Example List Tags:

```
{
  "tags": ["tag1", "tag2"]
}
```

PUT

/servers/{server_id}/tags

Replace Tags

Replaces all tags on specified server with the new set of tags.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-------------------------------------------------------------------------------------------------|
| server_id | path | string | The UUID of the server. |
| tags | body | array | A list of tags. The maximum count of tags in this list is 50. New in version 2.26 |

Example Replace Tags:

```
{
  "tags": ["tag1", "tag2"]
}
```

Response

| Name | In | Type | Description |
|------|------|-------|-------------------------------------------------------------------------------------------------|
| tags | body | array | A list of tags. The maximum count of tags in this list is 50. New in version 2.26 |

Example Replace Tags:

```
{
  "tags": ["tag1", "tag2"]
}
```

DELETE

/servers/{server_id}/tags

Delete All Tags

Deletes all tags from the specified server.

Normal response codes: 204

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

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

GET

/servers/{server_id}/tags/{tag}

Check Tag Existence

Checks tag existence on the server. If tag exists response with 204 status code will be returned. Otherwise returns 404.

Normal response codes: 204

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

Request

| Name | In | Type | Description |
|-----------|------|--------|--------------------------------------------------------|
| server_id | path | string | The UUID of the server. |
| tag | path | string | The tag as a string. New in version 2.26 |

PUT

/servers/{server_id}/tags/{tag}

Add a Single Tag

Adds a single tag to the server if server has no specified tag. Response code in this case is 201.

If the server has specified tag just returns 204.

Normal response codes: 201, 204

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

Request

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

| Name | In | Type | Description |
|------|------|--------|----------------------------------------------------|
| tag | path | string | The tag as a string. New in version 2.26 |

Response

| Name | In | Type | Description |
|----------|--------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Location | header | string | The location of the tag. It's individual tag URL which can be used for checking the existence of the tag on the server or deleting the tag from the server. New in version 2.26 |

DELETE

/servers/{server_id}/tags/{tag}

Delete a Single Tag

Deletes a single tag from the specified server.

Normal response codes: 204

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

Request

| Name | In | Type | Description |
|-----------|------|--------|----------------------------------------------------|
| server_id | path | string | The UUID of the server. |
| tag | path | string | The tag as a string. New in version 2.26 |

Response

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

Compute services (os-services)

Lists all running Compute services for a tenant, enables or disables scheduling for a Compute service, logs disabled Compute service information, and deletes a Compute service.

For an overview of Compute services, see [OpenStack Compute](#).

GET

/os-services

List Compute Services

Lists all running Compute services.

Provides details why any services were disabled.

Normal response codes: 200

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

Response

| Name | In | Type | Description |
|-----------------|------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| services | body | array | A list of service objects. |
| id | body | integer | The id of the service. |
| binary | body | string | The binary name of the service. |
| disabled_reason | body | string | The reason for disabling a service. |
| host | body | string | The name of the host. |
| state | body | string | The state of the service. One of up or down. |
| status | body | string | The status of the service. One of enabled or disabled. |
| updated_at | body | string | <p>The date and time when the resource was updated. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| forced_down | body | boolean | <p>Whether or not this service was forced down manually by an administrator. This value is useful to know that some 3rd party has verified the service should be marked down.</p> <p>New in version 2.11</p> |
| zone | body | string | The availability zone name. |

Example List Compute Services

```
{
  "services": [
    {
      "id": 1,
      "binary": "nova-scheduler",
      "disabled_reason": "test1",
      "host": "host1",
      "state": "up",

```

```

        "status": "disabled",
        "updated_at": "2012-10-29T13:42:02.000000",
        "forced_down": false,
        "zone": "internal"
    },
    {
        "id": 2,
        "binary": "nova-compute",
        "disabled_reason": "test2",
        "host": "host1",
        "state": "up",
        "status": "disabled",
        "updated_at": "2012-10-29T13:42:05.000000",
        "forced_down": false,
        "zone": "nova"
    },
    {
        "id": 3,
        "binary": "nova-scheduler",
        "disabled_reason": null,
        "host": "host2",
        "state": "down",
        "status": "enabled",
        "updated_at": "2012-09-19T06:55:34.000000",
        "forced_down": false,
        "zone": "internal"
    },
    {
        "id": 4,
        "binary": "nova-compute",
        "disabled_reason": "test4",
        "host": "host2",
        "state": "down",
        "status": "disabled",
        "updated_at": "2012-09-18T08:03:38.000000",
        "forced_down": false,
        "zone": "nova"
    }
]
}

```

PUT

/os-services/disable

Disable Scheduling For A Compute Service

Disables scheduling for a Compute service with optional logging.

Specify the service by its host name and binary name.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------|------|--------|---------------------------------|
| host | body | string | The name of the host. |
| binary | body | string | The binary name of the service. |

Example Disable Scheduling For A Compute Service

```
{
  "host": "host1",
  "binary": "nova-compute"
}
```

Response

| Name | In | Type | Description |
|---------|------|--------|-----------------------------------------------------------------------|
| service | body | object | Object representing a compute service. |
| binary | body | string | The binary name of the service. |
| host | body | string | The name of the host. |
| status | body | string | The status of the service. One of enabled or disabled . |

Example Disable Scheduling For A Compute Service

```
{
  "service": {
    "binary": "nova-compute",
    "host": "host1",
    "status": "disabled"
  }
}
```

PUT

/os-services/disable-log-reason

Log Disabled Compute Service Information

Logs information to the Compute service table about why a Compute service was disabled.

Specify the service by its host name and binary name.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------------|------|--------|-------------------------------------|
| host | body | string | The name of the host. |
| binary | body | string | The binary name of the service. |
| disabled_reason | body | string | The reason for disabling a service. |

Example Log Disabled Compute Service Information

```
{
```

```
"host": "host1",
"binary": "nova-compute",
"disabled_reason": "test2"
}
```

Response

| Name | In | Type | Description |
|-----------------|------|--------|-----------------------------------------------------------------------|
| service | body | object | Object representing a compute service. |
| binary | body | string | The binary name of the service. |
| disabled_reason | body | string | The reason for disabling a service. |
| host | body | string | The name of the host. |
| status | body | string | The status of the service. One of enabled or disabled . |

Example Log Disabled Compute Service Information

```
{
  "service": {
    "binary": "nova-compute",
    "disabled_reason": "test2",
    "host": "host1",
    "status": "disabled"
  }
}
```

PUT

/os-services/enable

Enable Scheduling For A Compute Service

Enables scheduling for a Compute service.

Specify the service by its host name and binary name.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------|------|--------|---------------------------------|
| host | body | string | The name of the host. |
| binary | body | string | The binary name of the service. |

Example Enable Scheduling For A Compute Service

```
{
  "host": "host1",
```

```
}  "binary": "nova-compute"
```

Response

| Name | In | Type | Description |
|---------|------|--------|-----------------------------------------------------------------------|
| service | body | object | Object representing a compute service. |
| binary | body | string | The binary name of the service. |
| host | body | string | The name of the host. |
| status | body | string | The status of the service. One of enabled or disabled . |

Example Enable Scheduling For A Compute Service

```
{  "service": {    "binary": "nova-compute",    "host": "host1",    "status": "enabled"  }}
```

PUT

/os-services/force-down

Update Forced Down

Set or unset `forced_down` flag for the service.

Action `force-down` available as of microversion 2.11.

Specify the service by its host name and binary name.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-------------|------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| host | body | string | The name of the host. |
| binary | body | string | The binary name of the service. |
| forced_down | body | boolean | Whether or not this service was forced down manually by an administrator. This value is useful to know that some 3rd party has verified the service should be marked down. |

| Name | In | Type | Description |
|------|----|------|----------------------------|
| | | | New in version 2.11 |

Example Update Forced Down

```
{
  "host": "host1",
  "binary": "nova-compute",
  "forced_down": true
}
```

Response

| Name | In | Type | Description |
|-------------|------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| service | body | object | Object representing a compute service. |
| binary | body | string | The binary name of the service. |
| forced_down | body | boolean | Whether or not this service was forced down manually by an administrator. This value is useful to know that some 3rd party has verified the service should be marked down. New in version 2.11 |
| host | body | string | The name of the host. |

Example Update Forced Down

```
{
  "service": {
    "binary": "nova-compute",
    "host": "host1",
    "forced_down": true
  }
}
```

DELETE

/os-services/{service_id}

Delete Compute Service

Deletes a Compute service.

Normal response codes: 204

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

Request

| Name | In | Type | Description |
|------------|------|---------|------------------------|
| service_id | path | integer | The id of the service. |

Response

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

Usage reports (os-simple-tenant-usage)

Reports usage statistics on compute and storage resources.

GET

/os-simple-tenant-usage

List Tenant Usage For All Tenants

Lists usage information for all tenants.

Normal response codes: 200

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

Response

Example List Tenant Usage For All Tenants: JSON response

```
{
  "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
    }
  ]
}
```

GET

/os-simple-tenant-usage/{tenant_id}

Show Usage Details For Tenant

Shows usage details 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. |

Response

Example Show Usage Details For Tenant: 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": "new-server-test",
        "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
  }
}
```

Create external events (os-server-external-events)

Warning

This is an admin level service API only designed to be used by other OpenStack services. The point of this API is to coordinate between Nova and Neutron (and potentially future services) on activities they both need to be involved in, such as network hotplugging.

Unless you are writing Neutron code you **should not** be using this API.

Creates one or more external events. The API dispatches each event to a server instance.

POST

/os-server-external-events

Run Events

Creates one or more external events, which the API dispatches to the host a server is assigned to. If the server is not currently assigned to a host the event will not be delivered.

You will receive back the list of events that you submitted, with an updated `code` and `status` indicating their level of success.

Normal response codes: 200, 207

A 200 will be returned if all events succeeded, 207 will be returned if some events could not be processed. The `code` attribute for the event will explain further what went wrong.

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

Request

| Name | In | Type | Description |
|-------------------|------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| events | body | array | The action. |
| name | body | string | The event name. A valid value is <code>network-changed</code> , <code>network-vif-plugged</code> , <code>network-vif-unplugged</code> , or <code>network-vif-deleted</code> . |
| server_uuid | body | string | The UUID of the server instance to which the API dispatches the event. You must assign this instance to a host. Otherwise, this call does not dispatch the event to the instance. |
| status (Optional) | body | string | The event status. A valid value is <code>failed</code> , <code>completed</code> , or <code>in-progress</code> . Default is <code>completed</code> . |
| tag (Optional) | body | string | A string value that identifies the event. |

Example Run Events

```
{
  "events": [
    {
      "name": "test-event",
      "tag": "foo",
      "status": "completed",
      "server_uuid": "3df201cf-2451-44f2-8d25-a4ca826fc1f3"
    }
  ]
}
```

Response

| Name | In | Type | Description |
|--------|------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| events | body | array | The action. |
| code | body | string | The HTTP response code for the event. The following codes are currently used: <ul style="list-style-type: none">200 - successfully submitted event404 - the instance specified by <code>server_uuid</code> was not found422 - no host was found for the server |

| Name | In | Type | Description |
|-------------------|------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | specified by <code>server_uuid</code> , so there is no route to this server. |
| name | body | string | The event name. A valid value is <code>network-changed</code> , <code>network-vif-plugged</code> , <code>network-vif-unplugged</code> , or <code>network-vif-deleted</code> . |
| server_uuid | body | string | The UUID of the server instance to which the API dispatches the event. You must assign this instance to a host. Otherwise, this call does not dispatch the event to the instance. |
| status (Optional) | body | string | The event status. A valid value is <code>failed</code> , <code>completed</code> , or <code>in-progress</code> . Default is <code>completed</code> . |
| tag (Optional) | body | string | A string value that identifies the event. |

Example Run Events

```
{
  "events": [
    {
      "code": 200,
      "name": "network-changed",
      "server_uuid": "ff1df7b2-6772-45fd-9326-c0a3b05591c2",
      "status": "completed",
      "tag": "foo"
    }
  ]
}
```

Extensions (extensions) (DEPRECATED)

Lists available extensions and shows information for an extension, by alias.

Nova originally supported the concept of API extensions, that allowed implementations of Nova to change the API (add new resources, or attributes to existing resource objects) via extensions. In an attempt to expose to the user what was supported in a particular site, the extensions resource provided a list of extensions and detailed information on each. The net result was gratuitous differentiation in the API that required all users of OpenStack clouds to write specific code to interact with every cloud.

As such, the entire extensions concept is deprecated, and will be removed in the near future.

For information about extensions, see [Extensions](#).

GET

/extensions

List Extensions

Lists all extensions to the API.

Normal response codes: 200

Error response codes: unauthorized(401)

Response

| Name | In | Type | Description |
|-------------|------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| extensions | body | array | List of extension objects. |
| name | body | string | Name of the extension. |
| alias | body | string | A short name by which this extension is also known. |
| links | body | array | Links pertaining to this extension. This is a list of dictionaries, each including keys <code>href</code> and <code>rel</code> . |
| namespace | body | string | A URL pointing to the namespace for this extension. |
| description | body | string | Text describing this extension's purpose. |
| 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. |

Example List Extensions

Lists all extensions to the API.

```
{
  "extensions": [
    {
      "alias": "NMN",
      "description": "Multiple network support.",
      "links": [],
      "name": "Multinic",
      "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
      "updated": "2014-12-03T00:00:00Z"
    },
    {
      "alias": "OS-DCF",
      "description": "Disk Management Extension.",
      "links": [],
      "name": "DiskConfig",
      "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
      "updated": "2014-12-03T00:00:00Z"
    },
    {
      "alias": "OS-EXT-AZ",
      "description": "Extended Availability Zone support.",
      "links": [],
      "name": "ExtendedAvailabilityZone",
      "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
      "updated": "2014-12-03T00:00:00Z"
    },
    {
      "alias": "OS-EXT-IMG-SIZE",
      "description": "Adds image size to image listings.",
      "links": [],
      "name": "ImageSize",
```

```

    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "OS-EXT-IPS",
    "description": "Adds type parameter to the ip list.",
    "links": [],
    "name": "ExtendedIps",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "OS-EXT-IPS-MAC",
    "description": "Adds mac address parameter to the ip list.",
    "links": [],
    "name": "ExtendedIpsMac",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "OS-EXT-SRV-ATTR",
    "description": "Extended Server Attributes support.",
    "links": [],
    "name": "ExtendedServerAttributes",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "OS-EXT-STS",
    "description": "Extended Status support.",
    "links": [],
    "name": "ExtendedStatus",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "OS-FLV-DISABLED",
    "description": "Support to show the disabled status of a flavor.",
    "links": [],
    "name": "FlavorDisabled",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "OS-FLV-EXT-DATA",
    "description": "Provide additional data for flavors.",
    "links": [],
    "name": "FlavorExtraData",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "OS-SCH-HNT",
    "description": "Pass arbitrary key/value pairs to the scheduler.",
    "links": [],
    "name": "SchedulerHints",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "OS-SRV-USG",
    "description": "Adds launched_at and terminated_at on Servers.",
    "links": [],
    "name": "ServerUsage",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-access-ips",
    "description": "Access IPs support.",
    "links": [],
    "name": "AccessIPs",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-admin-actions",
    "description": "Enable admin-only server actions\n\n    Actions include: resetNetwork,
injectNetworkInfo, os-resetState\n    ",
    "links": [],
    "name": "AdminActions",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-admin-password",

```

```

    "description": "Admin password management support.",
    "links": [],
    "name": "AdminPassword",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-agents",
    "description": "Agents support.",
    "links": [],
    "name": "Agents",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-aggregates",
    "description": "Admin-only aggregate administration.",
    "links": [],
    "name": "Aggregates",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-assisted-volume-snapshots",
    "description": "Assisted volume snapshots.",
    "links": [],
    "name": "AssistedVolumeSnapshots",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-attach-interfaces",
    "description": "Attach interface support.",
    "links": [],
    "name": "AttachInterfaces",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-availability-zone",
    "description": "1. Add availability_zone to the Create Server API.\n
zones describing.\n      ",
    "links": [],
    "name": "AvailabilityZone",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-baremetal-ext-status",
    "description": "Add extended status in Baremetal Nodes v2 API.",
    "links": [],
    "name": "BareMetalExtStatus",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-baremetal-nodes",
    "description": "Admin-only bare-metal node administration.",
    "links": [],
    "name": "BareMetalNodes",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-block-device-mapping",
    "description": "Block device mapping boot support.",
    "links": [],
    "name": "BlockDeviceMapping",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-block-device-mapping-v2-boot",
    "description": "Allow boot with the new BDM data format.",
    "links": [],
    "name": "BlockDeviceMappingV2Boot",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-cell-capacities",
    "description": "Adding functionality to get cell capacities.",
    "links": [],
    "name": "CellCapacities",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  }

```

```

    },
    {
        "alias": "os-cells",
        "description": "Enables cells-related functionality such as adding neighbor cells,\n
listing neighbor cells, and getting the capabilities of the local cell.\n    ",
        "links": [],
        "name": "Cells",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-certificates",
        "description": "Certificates support.",
        "links": [],
        "name": "Certificates",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-cloudpipe",
        "description": "Adds actions to create cloudpipe instances.\n\n    When running with the Vlan
network mode, you need a mechanism to route\n    from the public Internet to your vlans. This mechanism
is known as a\n    cloudpipe.\n\n    At the time of creating this class, only OpenVPN is supported.
Support for\n    a SSH Bastion host is forthcoming.\n    ",
        "links": [],
        "name": "Cloudpipe",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-cloudpipe-update",
        "description": "Adds the ability to set the vpn ip/port for cloudpipe instances.",
        "links": [],
        "name": "CloudpipeUpdate",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-config-drive",
        "description": "Config Drive Extension.",
        "links": [],
        "name": "ConfigDrive",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-console-auth-tokens",
        "description": "Console token authentication support.",
        "links": [],
        "name": "ConsoleAuthTokens",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-console-output",
        "description": "Console log output support, with tailing ability.",
        "links": [],
        "name": "ConsoleOutput",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-consoles",
        "description": "Interactive Console support.",
        "links": [],
        "name": "Consoles",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-create-backup",
        "description": "Create a backup of a server.",
        "links": [],
        "name": "CreateBackup",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-create-server-ext",
        "description": "Extended support to the Create Server v1.1 API.",
        "links": [],
        "name": "Createserverext",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {

```



```

        "alias": "os-deferred-delete",
        "description": "Instance deferred delete.",
        "links": [],
        "name": "DeferredDelete",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-evacuate",
        "description": "Enables server evacuation.",
        "links": [],
        "name": "Evacuate",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-extended-evacuate-find-host",
        "description": "Enables server evacuation without target host. Scheduler will select one to
target.",
        "links": [],
        "name": "ExtendedEvacuateFindHost",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-extended-floating-ips",
        "description": "Adds optional fixed_address to the add floating IP command.",
        "links": [],
        "name": "ExtendedFloatingIps",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-extended-hypervisors",
        "description": "Extended hypervisors support.",
        "links": [],
        "name": "ExtendedHypervisors",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-extended-networks",
        "description": "Adds additional fields to networks.",
        "links": [],
        "name": "ExtendedNetworks",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-extended-quotas",
        "description": "Adds ability for admins to delete quota and optionally force the update Quota
command.",
        "links": [],
        "name": "ExtendedQuotas",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-extended-rescue-with-image",
        "description": "Allow the user to specify the image to use for rescue.",
        "links": [],
        "name": "ExtendedRescueWithImage",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-extended-services",
        "description": "Extended services support.",
        "links": [],
        "name": "ExtendedServices",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-extended-services-delete",
        "description": "Extended services deletion support.",
        "links": [],
        "name": "ExtendedServicesDelete",
        "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
        "updated": "2014-12-03T00:00:00Z"
    },
    {
        "alias": "os-extended-status",
        "description": "Extended Status support.",
        "links": [],
        "name": "ExtendedStatus",

```

```

    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-extended-volumes",
    "description": "Extended Volumes support.",
    "links": [],
    "name": "ExtendedVolumes",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-fixed-ips",
    "description": "Fixed IPs support.",
    "links": [],
    "name": "FixedIPs",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-flavor-access",
    "description": "Flavor access support.",
    "links": [],
    "name": "FlavorAccess",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-flavor-extra-specs",
    "description": "Flavors extra specs support.",
    "links": [],
    "name": "FlavorExtraSpecs",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-flavor-manage",
    "description": "Flavor create/delete API support.",
    "links": [],
    "name": "FlavorManage",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-flavor-rxtx",
    "description": "Support to show the rxtx status of a flavor.",
    "links": [],
    "name": "FlavorRxtx",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-flavor-swap",
    "description": "Support to show the swap status of a flavor.",
    "links": [],
    "name": "FlavorSwap",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-floating-ip-dns",
    "description": "Floating IP DNS support.",
    "links": [],
    "name": "FloatingIpDns",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-floating-ip-pools",
    "description": "Floating IPs support.",
    "links": [],
    "name": "FloatingIpPools",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-floating-ips",
    "description": "Floating IPs support.",
    "links": [],
    "name": "FloatingIps",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-floating-ips-bulk",
    "description": "Bulk handling of Floating IPs.",

```

```

    "links": [],
    "name": "FloatingIpsBulk",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-fping",
    "description": "Fping Management Extension.",
    "links": [],
    "name": "Fping",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-hide-server-addresses",
    "description": "Support hiding server addresses in certain states.",
    "links": [],
    "name": "HideServerAddresses",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-hosts",
    "description": "Admin-only host administration.",
    "links": [],
    "name": "Hosts",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-hypervisor-status",
    "description": "Show hypervisor status.",
    "links": [],
    "name": "HypervisorStatus",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-hypervisors",
    "description": "Admin-only hypervisor administration.",
    "links": [],
    "name": "Hypervisors",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-instance-actions",
    "description": "View a log of actions and events taken on an instance.",
    "links": [],
    "name": "InstanceActions",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-instance_usage_audit_log",
    "description": "Admin-only Task Log Monitoring.",
    "links": [],
    "name": "OSInstanceUsageAuditLog",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-keypairs",
    "description": "Keypair Support.",
    "links": [],
    "name": "Keypairs",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-lock-server",
    "description": "Enable lock/unlock server actions.",
    "links": [],
    "name": "LockServer",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-migrate-server",
    "description": "Enable migrate and live-migrate server actions.",
    "links": [],
    "name": "MigrateServer",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {

```

```

    "alias": "os-migrations",
    "description": "Provide data on migrations.",
    "links": [],
    "name": "Migrations",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-multiple-create",
    "description": "Allow multiple create in the Create Server v2.1 API.",
    "links": [],
    "name": "MultipleCreate",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-networks",
    "description": "Admin-only Network Management Extension.",
    "links": [],
    "name": "Networks",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-networks-associate",
    "description": "Network association support.",
    "links": [],
    "name": "NetworkAssociationSupport",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-pause-server",
    "description": "Enable pause/unpause server actions.",
    "links": [],
    "name": "PauseServer",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-personality",
    "description": "Personality support.",
    "links": [],
    "name": "Personality",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-preserve-ephemeral-rebuild",
    "description": "Allow preservation of the ephemeral partition on rebuild.",
    "links": [],
    "name": "PreserveEphemeralOnRebuild",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-quota-class-sets",
    "description": "Quota classes management support.",
    "links": [],
    "name": "QuotaClasses",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-quota-sets",
    "description": "Quotas management support.",
    "links": [],
    "name": "Quotas",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-rescue",
    "description": "Instance rescue mode.",
    "links": [],
    "name": "Rescue",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-security-group-default-rules",
    "description": "Default rules for security group support.",
    "links": [],
    "name": "SecurityGroupDefaultRules",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  }

```

```

},
{
  "alias": "os-security-groups",
  "description": "Security group support.",
  "links": [],
  "name": "SecurityGroups",
  "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
  "updated": "2014-12-03T00:00:00Z"
},
{
  "alias": "os-server-diagnostics",
  "description": "Allow Admins to view server diagnostics through server action.",
  "links": [],
  "name": "ServerDiagnostics",
  "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
  "updated": "2014-12-03T00:00:00Z"
},
{
  "alias": "os-server-external-events",
  "description": "Server External Event Triggers.",
  "links": [],
  "name": "ServerExternalEvents",
  "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
  "updated": "2014-12-03T00:00:00Z"
},
{
  "alias": "os-server-group-quotas",
  "description": "Adds quota support to server groups.",
  "links": [],
  "name": "ServerGroupQuotas",
  "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
  "updated": "2014-12-03T00:00:00Z"
},
{
  "alias": "os-server-groups",
  "description": "Server group support.",
  "links": [],
  "name": "ServerGroups",
  "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
  "updated": "2014-12-03T00:00:00Z"
},
{
  "alias": "os-server-list-multi-status",
  "description": "Allow to filter the servers by a set of status values.",
  "links": [],
  "name": "ServerListMultiStatus",
  "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
  "updated": "2014-12-03T00:00:00Z"
},
{
  "alias": "os-server-password",
  "description": "Server password support.",
  "links": [],
  "name": "ServerPassword",
  "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
  "updated": "2014-12-03T00:00:00Z"
},
{
  "alias": "os-server-sort-keys",
  "description": "Add sorting support in get Server v2 API.",
  "links": [],
  "name": "ServerSortKeys",
  "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
  "updated": "2014-12-03T00:00:00Z"
},
{
  "alias": "os-server-start-stop",
  "description": "Start/Stop instance compute API support.",
  "links": [],
  "name": "ServerStartStop",
  "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
  "updated": "2014-12-03T00:00:00Z"
},
{
  "alias": "os-services",
  "description": "Services support.",
  "links": [],
  "name": "Services",
  "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
  "updated": "2014-12-03T00:00:00Z"
},
{
  "alias": "os-shelve",
  "description": "Instance shelve mode.",
  "links": [],
  "name": "Shelve",

```

```

    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-simple-tenant-usage",
    "description": "Simple tenant usage extension.",
    "links": [],
    "name": "SimpleTenantUsage",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-suspend-server",
    "description": "Enable suspend/resume server actions.",
    "links": [],
    "name": "SuspendServer",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-tenant-networks",
    "description": "Tenant-based Network Management Extension.",
    "links": [],
    "name": "OSTenantNetworks",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-used-limits",
    "description": "Provide data on limited resources that are being used.",
    "links": [],
    "name": "UsedLimits",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-used-limits-for-admin",
    "description": "Provide data to admin on limited resources used by other tenants.",
    "links": [],
    "name": "UsedLimitsForAdmin",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-user-data",
    "description": "Add user_data to the Create Server API.",
    "links": [],
    "name": "UserData",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-user-quotas",
    "description": "Project user quota support.",
    "links": [],
    "name": "UserQuotas",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-virtual-interfaces",
    "description": "Virtual interface support.",
    "links": [],
    "name": "VirtualInterfaces",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-volume-attachment-update",
    "description": "Support for updating a volume attachment.",
    "links": [],
    "name": "VolumeAttachmentUpdate",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  },
  {
    "alias": "os-volumes",
    "description": "Volumes support.",
    "links": [],
    "name": "Volumes",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  }
}

```

GET

/extensions/{alias}

Show Extension Details

Shows details for an extension, by alias.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-------|------|--------|-----------------------------------------------------|
| alias | body | string | A short name by which this extension is also known. |

Response

| Name | In | Type | Description |
|-------------|------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| extension | body | object | An extension object. |
| name | body | string | Name of the extension. |
| alias | body | string | A short name by which this extension is also known. |
| links | body | array | Links pertaining to this extension. This is a list of dictionaries, each including keys <code>href</code> and <code>rel</code> . |
| namespace | body | string | A URL pointing to the namespace for this extension. |
| description | body | string | Text describing this extension's purpose. |
| 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. |

Example Show Extension Details

Shows details about the `os-agents` extension.

```
{
  "extension": {
    "alias": "os-agents",
    "description": "Agents support.",
    "links": [],
    "name": "Agents",
    "namespace": "http://docs.openstack.org/compute/ext/fake_xml",
    "updated": "2014-12-03T00:00:00Z"
  }
}
```

}

Root certificates (os-certificates) (DEPRECATED)

Creates and shows details for a root certificate.

Warning

This API existed solely because of the need to build euca bundles when Nova had an in tree EC2 API. It no longer interacts with any parts of the system besides it's own certificate daemon. It is deprecated and will be removed in the near future.

POST

/os-certificates

Create Root Certificate

Creates a root certificate.

Normal response codes: 200

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

Response

| Name | In | Type | Description |
|-------------|------|--------|-------------------------|
| certificate | body | object | The certificate object. |
| data | body | string | The certificate. |
| private_key | body | string | The secret key. |

Example Create Root Certificate

```
{
  "certificate": {
    "data": "Certificate:\n      Data:\n      Version: 1 (0x0)\n      Serial Number: 1018 (0x3fa)\n      Signature Algorithm: md5WithRSAEncryption\n      Issuer: O=NOVA ROOT, L=Mountain View, ST=California, C=US\n      Validity\n      Not Before: Aug 12 07:20:30 2013 GMT\n      Not After : Aug 12 07:20:30 2014 GMT\n      Subject: C=US, ST=California, O=OpenStack, OU=NovaDev, CN=openstack-fake-2013-08-12T07:20:30Z\n      Subject Public Key Info:\n      Public Key Algorithm: rsaEncryption\n      Public-Key: (1024 bit)\n      Modulus:\n      00:ac:ff:b1:d1:ed:54:4e:35:6c:34:b4:8f:0b:04:\n      50:25:a3:e2:4f:02:4c:4f:26:59:bd:f3:fd:eb:da:\n      18:c2:36:aa:63:42:72:1f:88:4f:3a:ec:e7:9f:8e:\n      44:2a:d3:b8:94:7b:20:41:f8:48:02:57:91:4c:16:\n      62:f1:21:d4:f2:40:b5:86:50:d9:61:f0:be:ff:d8:\n      8d:9f:4b:aa:6a:07:38:a2:7f:87:21:fc:e6:6e:1d:\n      0a:95:1a:90:0e:60:c2:24:e9:8e:e8:68:1b:e9:f3:\n      c6:b0:7c:da:c5:20:66:9b:85:ea:f5:c9:a7:de:ee:\n      16:b1:51:a0:4d:e3:95:98:df\n      Exponent: 65537 (0x10001)\n      Signature
```



```
Algorithm: md5WithRSAEncryption\n
15:42:ca:71:cc:32:af:dc:cf:45:91:df:8a:b8:30:c4:7f:78:\n
80:a7:25:c2:d9:81:3e:b3:dd:22:cc:3b:f8:94:e7:8f:04:f6:\n
93:04:9e:85:d4:10:40:ff:5a:07:47:24:b5:ae:93:ad:8d:e1:\n
e6:54:4a:8d:4a:29:53:c4:8d:04:6b:0b:f6:af:38:78:02:c5:\n
05:19:89:82:2d:ba:fd:11:3c:1e:18:c9:0c:3d:03:93:6e:bc:\n
66:70:34:ee:03:78:8a:1d:3d:64:e8:20:2f:90:81:8e:49:1d:\n
07:37:15:66:42:cb:58:39:ad:56:ce:ed:47:c6:78:0b:0e:75:\n
29:ca\n-----BEGIN
CERTIFICATE-----\nMIICNDCCA0AgP6MA0GCSqGSIb3DQEBAUAME4xEjAQBgnVBAoTCU5PVKEgUk9P\nnVDEWMBQGA1
UEBxMNTW91bnRhaw4gVmlldzETMBEGA1UECBMKQ2FsaWZvcn5pYTEL\nnMAKGA1UEBhMCVVMwHhcNMTMwODEyMDcyMDMwWh
cNMTQwODEyMDcyMDMwWjB2MQsw\nnCQYDVQGEwJVUzETMBEGA1UECAwKQ2FsaWZvcn5pYTESMBAGA1UECgwJT3B1b1N0\nn
YWNrMRAwDgYDVQQLDAdOb3ZhrGV2MSwwKgYDVQQDDCNvcGVuc3RhY2stZmFrZS0y\nnMDEzLTA4LTEyVDA3OjIwOjMwWjCB
nzANBgkqhkiG9w0BAQEFAA0BjQAwgYkCgYEA\nnrP+x0e1UTjVsNLSPCwRQJaPiTwJMTyZZvfP969oYwjaqY0JyH4hP0uzn
n45Ekt04\nn1HsgQfhIAleRTBZi8SHU8kC1hLDZYfC+/9iNn0uqagc4on+HIfzmbh0K1RqQDmDC\nnJ0m06Ggb6fPGsHzaxS
Bmm4Xq9cmn3u4WsVGgTeOvMn8CAwEAATANBgkqhkiG9w0B\nnAQQAFAA0BgQAVQspxzDKv3M9Fkd+KuDDEf3iApyXC2YE+s9
0izDv4l0ePBPATBJ6F\nn1BBA/1oHRYs1rp0tjeHmVEqNSilTxi0Eawv2rzh4AsUFGYmCLbr9ETweGMkMPQOT\nnbrxmcDTu
A3iKHT1k6CAvKIGOSR0HNxVmQstY0a1Wzu1HxngLDnUpyg==\n-----END CERTIFICATE-----\n",
    "private_key": "-----BEGIN RSA PRIVATE
KEY-----\nMIICXgIBAAKBgQCs/7HR7VRONWw0tI8LBFAlo+JPAkxPJlm98/3r2hjCNqpjQnIf\n\nniE8670efjkQq07iUey
BB+EgCV5FMFmLxIdTyQLWGUNlh8L7/2I2fS6pqBzii4ch\n\n/OZuHQqVGpA0YMIk6Y7oaBvp88awfNrFIGabher1yafe7h
axUaBN45WY3wIDAQAB\n\nAoGBAIRcr2I/Kywf0hw4Nn10V9TuyE/9Gz2JHg3QFKjFJox2DqygADT5WAeHc6Bq\n\nnNKNf0NA2
SL1LSpm+ql01tv0w4VjE5TF60HiIzHuTnXggG6vuA8rxp6L24HtkAcC\n\nn0CBno9ggSX6jVornJPBfxpkwITYsvH57BUFV
D7ovbPyWGzS5AkeA1JeUtl6zxwps\n\nnWRr1aJ8I1l2uQk/RUIvSZOU61s+B190zvHikFy8LD8CI6vvBmjC/IZuZVedufjqs
\n4vX82uD03QJBANBSH2b2dyB4AGVfy9vXMRtALAspJHbLHy+zTKx1GPFiuz7Se3ps\n\n\nn8Kehz4C/CBXgQkk194dwFSGE19
/PQfyJROsCQCFFDJZhrTBUMwMZ2zSRiN5BUGt\n\n\nnbwuncS+0S1Su3Yz5VRYq2BZYEPHktYrAfKLWQ8eRwTawan5pFE/fb3
80gQXdAka4\n\nnDm0W/K0z1HbuyUxEpNQ28/6mBi0ktiWLT0tioq6sYmXLwZA/D2JrhXrG/xt/ol3\n\n\nnr8jqrfrNRsLByLhAg
h0N/AkeA12eR00971TEgFNqzIQwVmIAN9mB03cnf3tYcvlDU\n\n\nnm6eb2CS242y4Qa1fCCAEjxoJURdfsm3/D1iFo00X+IWF
+A==\n-----END RSA PRIVATE KEY-----\n"
  }
}
```

GET

/os-certificates/root

Show Root Certificate Details

Shows details for a root certificate.

Normal response codes: 200

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

Response

| Name | In | Type | Description |
|-------------|------|--------|-------------------------|
| certificate | body | object | The certificate object. |
| data | body | string | The certificate. |
| private_key | body | string | The secret key. |

Example Show Root Certificate Details

```
{
  "certificate": {
    "data": "-----BEGIN
CERTIFICATE-----\nMIICyZCCAjsGAwIBAgIJAj8zSiXup/m4MA0GCSqGSIb3DQEBAUAME4xEjAQBgnV\n\nBAoTCU5PVK
EgUk9PVDEWMBQGA1UEBxMNTW91bnRhaw4gVmlldzETMBEGA1UECBMK\n\nnQ2FsaWZvcn5pYTELMAKGA1UEBhMCVVMwHhcNMT
IxMDEzMDEzMzUwHcNMTMxMDE3\n\n\nnMDEzMzUwWjBOMRIwEAYDVQQKEw1OT1ZBIFJPT1Qx\n\n\nFjAUBgnVBACnTDU1vdW50Ywlu\n\n\nIFZpZCxcEzARBgnVBAGTCkNhbg1mb3JuawExCzAJBgNVBAYTA1VTMIGFMA0GCSqG\n\n\n\nnSiB3DQEBAQUAA4GNADCBiQKBgQDX
W4QfQqXJG4MqurqK8nU/Lge0mfNKxXj/Gwvg\n\n\nn2sQVwxzmKfoxih8Nn6yt0yHMNjhoji1UoWI03TXUnPZRAZmsypGKZeBd
7Y1Z0CPB\n\n\nnXGZVGrQm+PB2kZU+3cD8fVKcueMLLeZ+LRT5d0njnoKhc5xjqM1fFPimHMba40L6\n\n\n\nnTnYzPQIDAQABo4GwMI
GtMAwGA1UdEwQFMAMBAf8wHQYDVR00BBYEFKyoKu4SM0FM\n\n\nngx5Ec7p0nrCkabvxMH4GA1UdIwR3MHwAFKyoKu4SM0FMgx
-----END
CERTIFICATE-----\n"
  }
}
```

```

5Ec7p0nrCkabvxoVkk\nUDBOMRIwEAYDVQKKEw1OT1ZBIFJPT1QxFjAUBgNVBAcTDU1vdW50YWluIFZpZXcx\nEzARBgNV
BAGTCkNhbgGmb3JuaWExCzAJBgNVBAYTA1VTggkAnzNIjFSn+bgwDQYJ\nKoZIHvcNAQEEBQADgYEAXuvXlu1o/SVvykSL
hHW8QiAY00yzN/eDzYmZGomgiuo0\n/x+ayVzbrz1UWZnBD+1C4h112iELSmf22LjLoF+s/9NyPqHxGL3FrfatBkndaiF8
\nAx/TMEyCP17IQWi+3zzatqOKHSHiG7a9SGn/7o2aNTIWKVu1fy5GvmbBjBM/0UE=\n-----END
CERTIFICATE-----\n",
    "private_key": null
  }
}

```

Networks (os-networks) (DEPRECATED)

Warning

The networks API was designed to work with `nova-network`. Some features are proxied to `neutron` when appropriate, but as with all translation proxies, this is far from perfect compatibility. These APIs should be avoided in new applications in favor of using `neutron` directly. These will fail with a 404 after microversion 2.35. See: [Relevant Network APIs](#).

Creates, lists, shows information for, and deletes networks.

Adds network to a project, disassociates a network from a project, and disassociates a project from a network.

Associates host with and disassociates host from a network.

GET

/os-networks

List Networks

Lists networks for the project.

Policy defaults enable all users 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

Example List Networks: JSON response

```

{
  "networks": [
    {
      "bridge": "br100",
      "bridge_interface": "eth0",
      "broadcast": "10.0.0.7",
      "cidr": "10.0.0.0/29",
      "cidr_v6": null,
      "created_at": "2011-08-15T06:19:19.387525",
      "deleted": false,
      "deleted_at": null,
      "dhcp_server": "10.0.0.1",
      "dhcp_start": "10.0.0.3",
      "dns1": null,

```

```

        "dns2": null,
        "enable_dhcp": true,
        "gateway": "10.0.0.1",
        "gateway_v6": null,
        "host": "nsokolov-desktop",
        "id": "20c8acc0-f747-4d71-a389-46d078ebf047",
        "injected": false,
        "label": "mynet_0",
        "mtu": null,
        "multi_host": false,
        "netmask": "255.255.255.248",
        "netmask_v6": null,
        "priority": null,
        "project_id": "6133f8b603924f45bc0c9e21f6df12fa",
        "rxtx_base": null,
        "share_address": false,
        "updated_at": "2011-08-16T09:26:13.048257",
        "vlan": 100,
        "vpn_private_address": "10.0.0.2",
        "vpn_public_address": "127.0.0.1",
        "vpn_public_port": 1000
    },
    {
        "bridge": "br101",
        "bridge_interface": "eth0",
        "broadcast": "10.0.0.15",
        "cidr": "10.0.0.10/29",
        "cidr_v6": null,
        "created_at": "2011-08-15T06:19:19.885495",
        "deleted": false,
        "deleted_at": null,
        "dhcp_server": "10.0.0.9",
        "dhcp_start": "10.0.0.11",
        "dns1": null,
        "dns2": null,
        "enable_dhcp": true,
        "gateway": "10.0.0.9",
        "gateway_v6": null,
        "host": null,
        "id": "20c8acc0-f747-4d71-a389-46d078ebf000",
        "injected": false,
        "label": "mynet_1",
        "mtu": null,
        "multi_host": false,
        "netmask": "255.255.255.248",
        "netmask_v6": null,
        "priority": null,
        "project_id": null,
        "rxtx_base": null,
        "share_address": false,
        "updated_at": null,
        "vlan": 101,
        "vpn_private_address": "10.0.0.10",
        "vpn_public_address": null,
        "vpn_public_port": 1001
    }
]
}

```

POST

/os-networks

Create Network

Creates a network.

Policy defaults enable only users with the administrative role or the owner of the network 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)`, `conflict(409)`, `NotImplemented(501)`

Request

Example Create Network: JSON request

```
{
  "network": {
    "label": "new net 111",
    "cidr": "10.20.105.0/24",
    "mtu": 9000,
    "dhcp_server": "10.20.105.2",
    "enable_dhcp": false,
    "share_address": true,
    "allowed_start": "10.20.105.10",
    "allowed_end": "10.20.105.200"
  }
}
```

Response

Example Create Network: JSON response

```
{
  "network": {
    "bridge": null,
    "bridge_interface": null,
    "broadcast": "10.20.105.255",
    "cidr": "10.20.105.0/24",
    "cidr_v6": null,
    "created_at": null,
    "deleted": null,
    "deleted_at": null,
    "dhcp_server": "10.20.105.2",
    "dhcp_start": "10.20.105.2",
    "dns1": null,
    "dns2": null,
    "enable_dhcp": false,
    "gateway": "10.20.105.1",
    "gateway_v6": null,
    "host": null,
    "id": "d7a17c0c-457e-4ab4-a99c-4fa1762f5359",
    "injected": null,
    "label": "new net 111",
    "mtu": 9000,
    "multi_host": null,
    "netmask": "255.255.255.0",
    "netmask_v6": null,
    "priority": null,
    "project_id": null,
    "rxtx_base": null,
    "share_address": true,
    "updated_at": null,
    "vlan": null,
    "vpn_private_address": null,
    "vpn_public_address": null,
    "vpn_public_port": null
  }
}
```

POST

/os-networks/add

Add Network

Adds a network to a project.

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: 202

Error response codes: `badRequest(400)`, `unauthorized(401)`, `forbidden(403)`, `NotImplemented(501)`

Request

Example Add Network: JSON request

```
{
  "id": "1"
}
```

Response

GET

/os-networks/{network_id}

Show Network Details

Shows details for a network.

Policy defaults enable all users 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)`, `itemNotFound(404)`

Request

| Name | In | Type | Description |
|------------|------|--------|--------------------------|
| network_id | path | string | The UUID of the network. |

Response

Example Show Network Details: JSON response

```
{
  "network": {
    "bridge": "br100",
    "bridge_interface": "eth0",
    "broadcast": "10.0.0.7",
    "cidr": "10.0.0.0/29",
    "cidr_v6": null,
    "created_at": "2011-08-15T06:19:19.387525",
    "deleted": false,
  }
}
```

```
"deleted_at": null,
"dhcp_server": "10.0.0.1",
"dhcp_start": "10.0.0.3",
"dns1": null,
"dns2": null,
"enable_dhcp": true,
"gateway": "10.0.0.1",
"gateway_v6": null,
"host": "nsokolov-desktop",
"id": "20c8acc0-f747-4d71-a389-46d078ebf047",
"injected": false,
"label": "mynet_0",
"mtu": null,
"multi_host": false,
"netmask": "255.255.255.248",
"netmask_v6": null,
"priority": null,
"project_id": "6133f8b603924f45bc0c9e21f6df12fa",
"rxtx_base": null,
"share_address": false,
"updated_at": "2011-08-16T09:26:13.048257",
"vlan": 100,
"vpn_private_address": "10.0.0.2",
"vpn_public_address": "127.0.0.1",
"vpn_public_port": 1000
}
}
```

DELETE

/os-networks/{network_id}

Delete Network

Deletes a network.

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

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|------------|------|--------|--------------------------|
| network_id | path | string | The UUID of the network. |

Response

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

POST

/os-networks/{network_id}/action

Associate Host (DEPRECATED)

Warning

This API is only available with nova-network which is deprecated. It should be avoided in any new applications.

Associates a network with a host.

Specify the `associate_host` action in the request body.

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

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|----------------|------|--------|------------------------------------|
| network_id | path | string | The UUID of the network. |
| associate_host | body | string | The name of the host to associate. |

Example Associate Host to Network: JSON request

```
{
  "associate_host": "testHost"
}
```

Response

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

POST

/os-networks/{network_id}/action

Disassociate Network (DEPRECATED)

Warning

This API is only available with nova-network which is deprecated. It should be avoided in any new applications.

Disassociates a network from a project. You can then reuse the network.

Specify the `disassociate` action in the request body.

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

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|------------|------|--------|--------------------------|
| network_id | path | string | The UUID of the network. |

Example Disassociate Network: JSON request

```
{
  "disassociate": null
}
```

Response

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

POST

/os-networks/{network_id}/action

Disassociate Host (DEPRECATED)

Warning

This API is only available with nova-network which is deprecated. It should be avoided in any new applications.

Disassociates a host from a network.

Specify the `disassociate_host` action in the request body.

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

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|------------|------|--------|--------------------------|
| network_id | path | string | The UUID of the network. |

Example Disassociate Host from Network: JSON request

```
{
  "disassociate_host": null
}
```

Response

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

POST

/os-networks/{network_id}/action

Disassociate Project (DEPRECATED)

Warning

This API is only available with `nova-network` which is deprecated. It should be avoided in any new applications.

Disassociates a project from a network.

Specify the `disassociate_project` action in the request body.

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

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|-------------------------|-------------------|---------------------|--------------------------|
| <code>network_id</code> | <code>path</code> | <code>string</code> | The UUID of the network. |

Example Disassociate Project from Network: JSON request

```
{
  "disassociate_project": null
}
```

Response

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

Volume extension (os-volumes, os-snapshots) (DEPRECATED)

Warning

These APIs are proxy calls to the Volume service. Nova has deprecated all the proxy APIs and users should use the native APIs instead. These will fail with a 404 after microversion 2.35. See: [Relevant Volume APIs](#).

Manages volumes and snapshots for use with the Compute API. Lists, shows details, creates, and deletes volumes and snapshots.

GET

`/os-volumes`

List Volumes

Lists the volumes associated with the account.

Normal response codes: 200

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

Response

| Name | In | Type | Description |
|---------------------------------|------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| volumes | body | array | The list of volume objects. |
| attachments | body | array | The list of volume attachments. |
| attachments.device (Optional) | body | string | Name of the device in the attachment object, such as, /dev/vdb. |
| attachments.id (Optional) | body | string | The UUID of the attachment. |
| attachments.serverId (Optional) | body | string | The UUID of the server. |
| attachments.volumeId (Optional) | body | string | The UUID of the attached volume. |
| availabilityZone | body | string | The availability zone name. |
| createdAt | 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. |
| displayDescription | body | string | The volume description. |
| displayName | body | string | The volume name. |
| id | body | string | The UUID of the volume. |
| metadata | body | object | Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each. |
| size | body | integer | The size of the volume, in gibibytes (GiB). |
| snapshotId | body | string | The UUID for a snapshot. |
| status | body | string | The status of the volume. |

| Name | In | Type | Description |
|------------|------|--------|--------------------------------------------------|
| volumeType | body | string | The name or unique identifier for a volume type. |

Example List Volumes

```
{
  "volumes": [
    {
      "attachments": [
        {
          "device": "/",
          "id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
          "serverId": "3912f2b4-c5ba-4aec-9165-872876fe202e",
          "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f803"
        }
      ],
      "availabilityZone": "zone1:host1",
      "createdAt": "2013-02-19T20:01:40.274897",
      "displayDescription": "Volume Description",
      "displayName": "Volume Name",
      "id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
      "metadata": {},
      "size": 100,
      "snapshotId": null,
      "status": "in-use",
      "volumeType": "Backup"
    }
  ]
}
```

POST

/os-volumes

Create Volume

Creates a new volume.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|------------------------------|------|---------|---------------------------------------------|
| volume | body | object | The volume object. |
| size | body | integer | The size of the volume, in gibibytes (GiB). |
| availability_zone (Optional) | body | string | The availability zone name. |
| display_name (Optional) | body | string | The volume name. |

| Name | In | Type | Description |
|--------------------------------|------|--------|-------------------------------------------------------------------------------------------------|
| display_description (Optional) | body | string | The volume description. |
| metadata (Optional) | body | object | Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each. |
| volume_type (Optional) | body | string | The unique identifier for a volume type. |
| snapshot_id (Optional) | body | string | The UUID for a snapshot. |

Example Create Volume

```
{
  "volume": {
    "availability_zone": "zone1:host1",
    "display_name": "Volume Name",
    "display_description": "Volume Description",
    "size": 100
  }
}
```

Response

| Name | In | Type | Description |
|---------------------------------|------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| volume | body | object | The volume object. |
| attachments | body | array | The list of volume attachments. |
| attachments.device (Optional) | body | string | Name of the device in the attachment object, such as, /dev/vdb. |
| attachments.id (Optional) | body | string | The UUID of the attachment. |
| attachments.serverId (Optional) | body | string | The UUID of the server. |
| attachments.volumeId (Optional) | body | string | The UUID of the attached volume. |
| availabilityZone | body | string | The availability zone name. |
| createdAt | body | string | <p>The date and time when the resource was created. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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</p> |

| Name | In | Type | Description |
|--------------------|------|---------|-------------------------------------------------------------------------------------------------|
| | | | offset value is -05:00. |
| displayName | body | string | The volume name. |
| displayDescription | body | string | The volume description. |
| id | body | string | The UUID of the volume. |
| metadata | body | object | Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each. |
| size | body | integer | The size of the volume, in gibibytes (GiB). |
| snapshotId | body | string | The UUID for a snapshot. |
| status | body | string | The status of the volume. |
| volumeType | body | string | The name or unique identifier for a volume type. |

Example Create Volume

```
{
  "volume": {
    "attachments": [
      {
        "device": "/",
        "id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
        "serverId": "3912f2b4-c5ba-4aec-9165-872876fe202e",
        "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f803"
      }
    ],
    "availabilityZone": "zone1:host1",
    "createdAt": "2013-02-18T14:51:17.970024",
    "displayDescription": "Volume Description",
    "displayName": "Volume Name",
    "id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
    "metadata": {},
    "size": 100,
    "snapshotId": null,
    "status": "in-use",
    "volumeType": "Backup"
  }
}
```

GET

/os-volumes/detail

List Volumes With Details

Lists all volumes with details.

Normal response codes: 200

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

Response

| Name | In | Type | Description |
|---------------------------------|------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| volumes | body | array | The list of volume objects. |
| attachments | body | array | The list of volume attachments. |
| attachments.device (Optional) | body | string | Name of the device in the attachment object, such as, /dev/vdb. |
| attachments.id (Optional) | body | string | The UUID of the attachment. |
| attachments.serverId (Optional) | body | string | The UUID of the server. |
| attachments.volumeId (Optional) | body | string | The UUID of the attached volume. |
| availabilityZone | body | string | The availability zone name. |
| createdAt | 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. |
| displayName | body | string | The volume name. |
| displayDescription | body | string | The volume description. |
| id | body | string | The UUID of the volume. |
| metadata | body | object | Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each. |
| size | body | integer | The size of the volume, in gibibytes (GiB). |
| snapshotId | body | string | The UUID for a snapshot. |
| status | body | string | The status of the volume. |
| volumeType | body | string | The name or unique identifier for a volume type. |

Example List Volumes With Details

{

```
"volumes": [
  {
    "attachments": [
      {
        "device": "/",
        "id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
        "serverId": "3912f2b4-c5ba-4aec-9165-872876fe202e",
        "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f803"
      }
    ],
    "availabilityZone": "zone1:host1",
    "createdAt": "1999-01-01T01:01:01.000000",
    "displayDescription": "Volume Description",
    "displayName": "Volume Name",
    "id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
    "metadata": {},
    "size": 100,
    "snapshotId": null,
    "status": "in-use",
    "volumeType": "Backup"
  }
]
```

GET

/os-volumes/{volume_id}

Show Volume Details

Shows details for a given volume.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-----------------------------|
| volume_id | path | string | The unique ID for a volume. |

Response

| Name | In | Type | Description |
|------------------------------------|------|--------|-----------------------------------------------------------------|
| volume | body | object | The volume object. |
| attachments | body | array | The list of volume attachments. |
| attachment.device (Optional) | body | string | Name of the device in the attachment object, such as, /dev/vdb. |
| attachments.id (Optional) | body | string | The UUID of the attachment. |
| attachments.serverId (Optional) | body | string | The UUID of the server. |
| attachments.volumeId | body | string | The UUID of the attached volume. |

| Name | In | Type | Description |
|--------------------|------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (Optional) | | | |
| availabilityZone | body | string | The availability zone name. |
| createdAt | body | string | <p>The date and time when the resource was created. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| displayName | body | string | The volume name. |
| displayDescription | body | string | The volume description. |
| id | body | string | The UUID of the volume. |
| metadata | body | object | Metadata key and value pairs. The maximum size of the metadata key and value is 255 bytes each. |
| size | body | integer | The size of the volume, in gibibytes (GiB). |
| snapshotId | body | string | The UUID for a snapshot. |
| status | body | string | The status of the volume. |
| volumeType | body | string | The name or unique identifier for a volume type. |

Example Show Volume Details

```
{
  "volume": {
    "attachments": [
      {
        "device": "/",
        "id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
        "serverId": "3912f2b4-c5ba-4aec-9165-872876fe202e",
        "volumeId": "a26887c6-c47b-4654-abb5-dfadf7d3f803"
      }
    ],
    "availabilityZone": "zone1:host1",
    "createdAt": "2013-02-18T14:51:18.528085",
    "displayDescription": "Volume Description",
    "displayName": "Volume Name",
    "id": "a26887c6-c47b-4654-abb5-dfadf7d3f803",
    "metadata": {},
    "size": 100,
    "snapshotId": null,
    "status": "in-use",
    "volumeType": "Backup"
  }
}
```



```
}  
}
```

DELETE

/os-volumes/{volume_id}

Delete Volume

Deletes a volume.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-----------------------------|
| volume_id | path | string | The unique ID for a volume. |

Response

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

GET

/os-snapshots

List Snapshots

Lists snapshots.

Normal response codes: 200

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

Response

| Name | In | Type | Description |
|--------------------|------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| snapshots | body | array | A list of snapshot objects. |
| id | body | string | The UUID for a snapshot. |
| createdAt | 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. |
| displayName | body | string | The volume name. |
| displayDescription | body | string | The volume description. |

| Name | In | Type | Description |
|----------|------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| size | body | integer | The size of the volume, in gibibytes (GiB). |
| status | body | string | The status of the snapshot. Valid status values are: <ul style="list-style-type: none"> • available • creating • deleting • error • error_deleting |
| volumeId | body | string | The source volume ID. |

Example List Snapshots

```
{
  "snapshots": [
    {
      "createdAt": "2013-02-25T16:27:54.684999",
      "displayDescription": "Default description",
      "displayName": "Default name",
      "id": 100,
      "size": 100,
      "status": "available",
      "volumeId": 12
    },
    {
      "createdAt": "2013-02-25T16:27:54.685005",
      "displayDescription": "Default description",
      "displayName": "Default name",
      "id": 101,
      "size": 100,
      "status": "available",
      "volumeId": 12
    },
    {
      "createdAt": "2013-02-25T16:27:54.685008",
      "displayDescription": "Default description",
      "displayName": "Default name",
      "id": 102,
      "size": 100,
      "status": "available",
      "volumeId": 12
    }
  ]
}
```

POST

/os-snapshots

Create Snapshot

Creates a new snapshot.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------------------------------|------|---------|---------------------------------------------------------------------------|
| snapshot | body | object | A partial representation of a snapshot that is used to create a snapshot. |
| volume_id | body | string | The source volume ID. |
| display_description (Optional) | body | string | The volume description. |
| display_name (Optional) | body | string | The volume name. |
| force (Optional) | body | boolean | Indicates whether to create a snapshot, even if the volume is attached. |

Example Create Snapshot

```
{
  "snapshot": {
    "display_name": "snap-001",
    "display_description": "Daily backup",
    "volume_id": "521752a6-acf6-4b2d-bc7a-119f9148cd8c",
    "force": false
  }
}
```

Response

| Name | In | Type | Description |
|--------------------|------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| snapshot | body | object | A partial representation of a snapshot that is used to create a snapshot. |
| id | body | string | The UUID for a snapshot. |
| createdAt | 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. |
| displayName | body | string | The volume name. |
| displayDescription | body | string | The volume description. |
| volumeId | body | string | The source volume ID. |

| Name | In | Type | Description |
|--------|------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| size | body | integer | The size of the volume, in gibibytes (GiB). |
| status | body | string | The status of the snapshot. Valid status values are: <ul style="list-style-type: none"> • available • creating • deleting • error • error_deleting |

Example Create Snapshot

```
{
  "snapshot": {
    "createdAt": "2013-02-25T16:27:54.680544",
    "displayDescription": "Daily backup",
    "displayName": "snap-001",
    "id": 100,
    "size": 100,
    "status": "available",
    "volumeId": "521752a6-acf6-4b2d-bc7a-119f9148cd8c"
  }
}
```

GET

/os-snapshots/detail

List Snapshots With Details

Lists all snapshots with details.

Normal response codes: 200

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

Response

| Name | In | Type | Description |
|--------------------|------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| snapshots | body | array | A list of snapshot objects. |
| id | body | string | The UUID for a snapshot. |
| createdAt | 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. |
| displayName | body | string | The volume name. |
| displayDescription | body | string | The volume description. |

| Name | In | Type | Description |
|----------|------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| volumeId | body | string | The source volume ID. |
| size | body | integer | The size of the volume, in gibibytes (GiB). |
| status | body | string | The status of the snapshot. Valid status values are: <ul style="list-style-type: none"> • available • creating • deleting • error • error_deleting |

Example List Snapshots With Details

```
{
  "snapshots": [
    {
      "createdAt": "2013-02-25T16:27:54.671372",
      "displayDescription": "Default description",
      "displayName": "Default name",
      "id": 100,
      "size": 100,
      "status": "available",
      "volumeId": 12
    },
    {
      "createdAt": "2013-02-25T16:27:54.671378",
      "displayDescription": "Default description",
      "displayName": "Default name",
      "id": 101,
      "size": 100,
      "status": "available",
      "volumeId": 12
    },
    {
      "createdAt": "2013-02-25T16:27:54.671381",
      "displayDescription": "Default description",
      "displayName": "Default name",
      "id": 102,
      "size": 100,
      "status": "available",
      "volumeId": 12
    }
  ]
}
```

GET

/os-snapshots/{snapshot_id}

Show Snapshot Details

Shows details for a given snapshot.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-------------|------|--------|---------------------------|
| snapshot_id | path | string | The UUID of the snapshot. |

Response

| Name | In | Type | Description |
|--------------------|------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| snapshot | body | object | A partial representation of a snapshot that is used to create a snapshot. |
| id | body | string | The UUID for a snapshot. |
| createdAt | 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. |
| displayName | body | string | The volume name. |
| displayDescription | body | string | The volume description. |
| volumeId | body | string | The source volume ID. |
| size | body | integer | The size of the volume, in gibibytes (GiB). |
| status | body | string | The status of the snapshot. Valid status values are: <ul style="list-style-type: none">• available• creating• deleting• error• error_deleting |

Example Show Snapshot Details

```
{
  "snapshot": {
    "createdAt": "2013-02-25T16:27:54.724209",
    "displayDescription": "Default description",
    "displayName": "Default name",
    "id": "100",
    "size": 100,
    "status": "available",
    "volumeId": 12
  }
}
```

DELETE

/os-snapshots/{snapshot_id}

Delete Snapshot

Deletes a snapshot from the account.

This operation is asynchronous. You must list snapshots repeatedly to determine whether the snapshot was deleted.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|-------------|------|--------|---------------------------|
| snapshot_id | path | string | The UUID of the snapshot. |

Response

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

Images (DEPRECATED)

Warning

These APIs are proxy calls to the Image service. Nova has deprecated all the proxy APIs and users should use the native APIs instead. These will fail with a 404 after microversion 2.35. See: [Relevant Image APIs](#).

Lists, shows details, creates, updates, and deletes images. Also sets, lists, shows details, create, update and deletes image metadata.

An image is a collection of files that you use to create and rebuild a server. By default, operators provide pre-built operating system images. You can also create custom images. See: [Create Image Action](#).

By default, the `policy.json` file authorizes all users to view the image size in the `OS-EXT-IMG-SIZE:size` extended attribute.

GET

/images

List Images

List images.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------------------------|-------|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| changes-since (Optional) | query | string | <p>Filters the response by a date and time when the image last changed status. Use this query parameter to check for changes since a previous request rather than re-downloading and re-parsing the full status at each polling interval. If data has changed, the call returns only the items changed since the <code>changes-since</code> time. If data has not changed since the <code>changes-since</code> time, the call returns an empty list. To enable you to keep track of changes, this filter also displays images that were deleted if the <code>changes-since</code> value specifies a date in the last 30 days. Items deleted more than 30 days ago might be returned, but it is not guaranteed. The date and time stamp format is ISO 8601:</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>The ±hh:mm value, if included, returns the time zone as an offset from UTC. For example, 2015-08-27T09:49:58-05:00. If you omit the time zone, the UTC time zone is assumed.</p> |
| server (Optional) | query | string | Filters the response by a server, as a URL. |
| name (Optional) | query | string | Filters the response by an image name, as a string. |
| status (Optional) | query | string | Filters the response by an image status, as a string. For example, <code>ACTIVE</code> . |
| 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 MB. For example, 512. |
| type (Optional) | query | string | Filters the response by an image type. For example, <code>snapshot</code> or <code>backup</code> . |
| limit (Optional) | query | integer | Requests a page size of items. Returns a number of items up to a limit value. Use the <code>limit</code> parameter to make an initial limited request and use the ID of the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request. |
| marker (Optional) | query | string | The ID of the last-seen item. Use the <code>limit</code> parameter to make an initial limited request and use the ID of the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request. |

Response

| Name | In | Type | Description |
|--------|------|--------|---------------------------------------------------------------------------------------------------------|
| images | body | array | An array of Image objects. |
| id | body | string | The ID of the Image. |
| name | body | string | The display name of an Image. |
| links | body | array | Links to the resources in question. See API Guide / Links and References for more info. |

Example List Images: JSON response

```
{
  "images": [
    {
      "id": "70a599e0-31e7-49b7-b260-868f441e862b",
      "links": [
        {
          "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/70a599e0-31e7-49b7-b260-868f441e862b",
          "rel": "self"
        },
        {
          "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/70a599e0-31e7-49b7-b260-868f441e862b",
          "rel": "bookmark"
        },
        {
          "href": "http://glance.openstack.example.com/images/70a599e0-31e7-49b7-b260-868f441e862b",
          "rel": "alternate",
          "type": "application/vnd.openstack.image"
        }
      ],
      "name": "fakeimage7"
    },
    {
      "id": "155d900f-4e14-4e4c-a73d-069cbf4541e6",
      "links": [
        {
          "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/155d900f-4e14-4e4c-a73d-069cbf4541e6",
          "rel": "self"
        },
        {
          "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/155d900f-4e14-4e4c-a73d-069cbf4541e6",
          "rel": "bookmark"
        },
        {
          "href": "http://glance.openstack.example.com/images/155d900f-4e14-4e4c-a73d-069cbf4541e6",
          "rel": "alternate",
          "type": "application/vnd.openstack.image"
        }
      ],
      "name": "fakeimage123456"
    },
    {
      "id": "a2459075-d96c-40d5-893e-577ff92e721c",
      "links": [
        {
          "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/a2459075-d96c-40d5-893e-577ff92e721c",
          "rel": "self"
        },
        {
          "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/a2459075-d96c-40d5-893e-577ff92e721c",
          "rel": "bookmark"
        },
        {
          "href": "http://glance.openstack.example.com/images/a2459075-d96c-40d5-893e-577ff92e721c",
          "rel": "alternate",
          "type": "application/vnd.openstack.image"
        }
      ],
      "name": "fakeimage123456"
    },
    {
      "id": "a440c04b-79fa-479c-bed1-0b816eae379",
      "links": [
        {
          "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/a440c04b-79fa-479c-bed1-0b816eae379",
```

```

        "rel": "self"
      },
      {
        "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/a440c04b-79fa-479c-bed1-0b816eae379",
        "rel": "bookmark"
      },
      {
        "href": "http://glance.openstack.example.com/images/a440c04b-79fa-479c-bed1-0b816eae379",
        "rel": "alternate",
        "type": "application/vnd.openstack.image"
      }
    ],
    "name": "fakeimage6"
  },
  {
    "id": "c905cedb-7281-47e4-8a62-f26bc5fc4c77",
    "links": [
      {
        "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/c905cedb-7281-47e4-8a62-f26bc5fc4c77",
        "rel": "self"
      },
      {
        "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/c905cedb-7281-47e4-8a62-f26bc5fc4c77",
        "rel": "bookmark"
      },
      {
        "href": "http://glance.openstack.example.com/images/c905cedb-7281-47e4-8a62-f26bc5fc4c77",
        "rel": "alternate",
        "type": "application/vnd.openstack.image"
      }
    ],
    "name": "fakeimage123456"
  },
  {
    "id": "cedef40a-ed67-4d10-800e-17455edce175",
    "links": [
      {
        "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/cedef40a-ed67-4d10-800e-17455edce175",
        "rel": "self"
      },
      {
        "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/cedef40a-ed67-4d10-800e-17455edce175",
        "rel": "bookmark"
      },
      {
        "href": "http://glance.openstack.example.com/images/cedef40a-ed67-4d10-800e-17455edce175",
        "rel": "alternate",
        "type": "application/vnd.openstack.image"
      }
    ],
    "name": "fakeimage123456"
  },
  {
    "id": "76fa36fc-c930-4bf3-8c8a-ea2a2420deb6",
    "links": [
      {
        "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/76fa36fc-c930-4bf3-8c8a-ea2a2420deb6",
        "rel": "self"
      },
      {
        "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/76fa36fc-c930-4bf3-8c8a-ea2a2420deb6",
        "rel": "bookmark"
      },
      {
        "href": "http://glance.openstack.example.com/images/76fa36fc-c930-4bf3-8c8a-ea2a2420deb6",
        "rel": "alternate",
        "type": "application/vnd.openstack.image"
      }
    ],
    "name": "fakeimage123456"
  }
]
}

```

GET

/images/detail

List Images With Details

List images with details.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------------------------|-------|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| changes-since (Optional) | query | string | <p>Filters the response by a date and time when the image last changed status. Use this query parameter to check for changes since a previous request rather than re-downloading and re-parsing the full status at each polling interval. If data has changed, the call returns only the items changed since the <code>changes-since</code> time. If data has not changed since the <code>changes-since</code> time, the call returns an empty list. To enable you to keep track of changes, this filter also displays images that were deleted if the <code>changes-since</code> value specifies a date in the last 30 days. Items deleted more than 30 days ago might be returned, but it is not guaranteed. The date and time stamp format is ISO 8601:</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>The ±hh:mm value, if included, returns the time zone as an offset from UTC. For example, 2015-08-27T09:49:58-05:00. If you omit the time zone, the UTC time zone is assumed.</p> |
| server (Optional) | query | string | Filters the response by a server, as a URL. |
| name (Optional) | query | string | Filters the response by an image name, as a string. |
| status (Optional) | query | string | Filters the response by an image status, as a string. For example, ACTIVE. |
| 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 MB. For example, 512. |
| type (Optional) | query | string | Filters the response by an image type. For example, snapshot or backup. |
| limit (Optional) | query | integer | Requests a page size of items. Returns a number of items up to a limit value. Use the <code>limit</code> parameter to make an initial limited request and use the ID of the last-seen item from the response as the <code>marker</code> parameter value in a subsequent limited request. |
| marker (Optional) | query | string | The ID of the last-seen item. Use the <code>limit</code> parameter to make an initial limited request and use the ID of the last-seen item from the response as the <code>marker</code> parameter |

| Name | In | Type | Description |
|------|----|------|----------------------------------------|
| | | | value in a subsequent limited request. |

Response

| Name | In | Type | Description |
|----------|------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| images | body | array | An array of Image objects. |
| id | body | string | The ID of the Image. |
| name | body | string | The display name of an Image. |
| minRam | body | integer | The minimum amount of RAM an image requires to function, in MB. For example, 512. |
| minDisk | body | integer | The minimum amount of disk space an image requires to boot, in GiB. For example, 100. |
| metadata | body | object | Metadata key and value pairs. The maximum size for each metadata key and value pair is 255 bytes. |
| 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. |
| 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. |
| status | body | string | The status of image, as a string. This can be one of: <ul style="list-style-type: none"> • ACTIVE: image is in active state • SAVING: image is in queued or in saving process • DELETED: image is deleted or in progress of deletion • ERROR: image is in error state • UNKNOWN: image is in unknown state |

| Name | In | Type | Description |
|------------------------------|------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| progress | body | integer | A percentage value of the image save progress. This can be one of: <ul style="list-style-type: none"> ACTIVE: 100 SAVING: 25 or 50 |
| links | body | array | Links to the resources in question. See API Guide / Links and References for more info. |
| server (Optional) | body | object | The server booted from image. |
| OS-EXT-IMG-SIZE:size | body | integer | The size of the image. |
| 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: <ul style="list-style-type: none"> 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. |

Example List Images Details: JSON response

```
{
  "images": [
    {
      "OS-DCF:diskConfig": "AUTO",
      "OS-EXT-IMG-SIZE:size": "74185822",
      "created": "2011-01-01T01:02:03Z",
      "id": "70a599e0-31e7-49b7-b260-868f441e862b",
      "links": [
        {
          "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/70a599e0-31e7-49b7-b260-868f441e862b",
          "rel": "self"
        },
        {
          "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/70a599e0-31e7-49b7-b260-868f441e862b",
          "rel": "bookmark"
        },
        {
          "href": "http://glance.openstack.example.com/images/70a599e0-31e7-49b7-b260-868f441e862b",
          "rel": "alternate",
          "type": "application/vnd.openstack.image"
        }
      ]
    },
    {
      "metadata": {
```

```

        "architecture": "x86_64",
        "auto_disk_config": "True",
        "kernel_id": "nokernel",
        "ramdisk_id": "nokernel"
    },
    "minDisk": 0,
    "minRam": 0,
    "name": "fakeimage7",
    "progress": 100,
    "status": "ACTIVE",
    "updated": "2011-01-01T01:02:03Z"
},
{
    "OS-EXT-IMG-SIZE:size": "74185822",
    "created": "2011-01-01T01:02:03Z",
    "id": "155d900f-4e14-4e4c-a73d-069cbf4541e6",
    "links": [
        {
            "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/155d900f-4e14-4e4c-a73d-069cbf4541e6",
            "rel": "self"
        },
        {
            "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/155d900f-4e14-4e4c-a73d-069cbf4541e6",
            "rel": "bookmark"
        },
        {
            "href": "http://glance.openstack.example.com/images/155d900f-4e14-4e4c-a73d-069cbf4541e6",
            "rel": "alternate",
            "type": "application/vnd.openstack.image"
        }
    ],
    "metadata": {
        "architecture": "x86_64",
        "kernel_id": "nokernel",
        "ramdisk_id": "nokernel"
    },
    "minDisk": 0,
    "minRam": 0,
    "name": "fakeimage123456",
    "progress": 100,
    "status": "ACTIVE",
    "updated": "2011-01-01T01:02:03Z"
},
{
    "OS-EXT-IMG-SIZE:size": "74185822",
    "created": "2011-01-01T01:02:03Z",
    "id": "a2459075-d96c-40d5-893e-577ff92e721c",
    "links": [
        {
            "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/a2459075-d96c-40d5-893e-577ff92e721c",
            "rel": "self"
        },
        {
            "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/a2459075-d96c-40d5-893e-577ff92e721c",
            "rel": "bookmark"
        },
        {
            "href": "http://glance.openstack.example.com/images/a2459075-d96c-40d5-893e-577ff92e721c",
            "rel": "alternate",
            "type": "application/vnd.openstack.image"
        }
    ],
    "metadata": {
        "kernel_id": "nokernel",
        "ramdisk_id": "nokernel"
    },
    "minDisk": 0,
    "minRam": 0,
    "name": "fakeimage123456",
    "progress": 100,
    "status": "ACTIVE",
    "updated": "2011-01-01T01:02:03Z"
},
{
    "OS-DCF:diskConfig": "MANUAL",
    "OS-EXT-IMG-SIZE:size": "74185822",
    "created": "2011-01-01T01:02:03Z",
    "id": "a440c04b-79fa-479c-bed1-0b816eaec379",
    "links": [
        {
            "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/a440c04b-79fa-479c-bed1-0b816eaec379",
            "rel": "self"
        },
        {
            "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/a440c04b-79fa-479c-bed1-0b816eaec379",
            "rel": "bookmark"
        },
        {
            "href": "http://glance.openstack.example.com/images/a440c04b-79fa-479c-bed1-0b816eaec379",
            "rel": "alternate",
            "type": "application/vnd.openstack.image"
        }
    ]
}

```

```

    ],
    "metadata": {
        "architecture": "x86_64",
        "auto_disk_config": "False",
        "kernel_id": "nokernel",
        "ramdisk_id": "nokernel"
    },
    "minDisk": 0,
    "minRam": 0,
    "name": "fakeimage6",
    "progress": 100,
    "status": "ACTIVE",
    "updated": "2011-01-01T01:02:03Z"
},
{
    "OS-EXT-IMG-SIZE:size": "74185822",
    "created": "2011-01-01T01:02:03Z",
    "id": "c905cedb-7281-47e4-8a62-f26bc5fc4c77",
    "links": [
        {
            "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/c905cedb-7281-47e4-8a62-f26bc5fc4c77",
            "rel": "self"
        },
        {
            "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/c905cedb-7281-47e4-8a62-f26bc5fc4c77",
            "rel": "bookmark"
        },
        {
            "href": "http://glance.openstack.example.com/images/c905cedb-7281-47e4-8a62-f26bc5fc4c77",
            "rel": "alternate",
            "type": "application/vnd.openstack.image"
        }
    ],
    "metadata": {
        "kernel_id": "155d900f-4e14-4e4c-a73d-069cbf4541e6",
        "ramdisk_id": null
    },
    "minDisk": 0,
    "minRam": 0,
    "name": "fakeimage123456",
    "progress": 100,
    "status": "ACTIVE",
    "updated": "2011-01-01T01:02:03Z"
},
{
    "OS-EXT-IMG-SIZE:size": "74185822",
    "created": "2011-01-01T01:02:03Z",
    "id": "cedef40a-ed67-4d10-800e-17455edce175",
    "links": [
        {
            "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/cedef40a-ed67-4d10-800e-17455edce175",
            "rel": "self"
        },
        {
            "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/cedef40a-ed67-4d10-800e-17455edce175",
            "rel": "bookmark"
        },
        {
            "href": "http://glance.openstack.example.com/images/cedef40a-ed67-4d10-800e-17455edce175",
            "rel": "alternate",
            "type": "application/vnd.openstack.image"
        }
    ],
    "metadata": {
        "kernel_id": "nokernel",
        "ramdisk_id": "nokernel"
    },
    "minDisk": 0,
    "minRam": 0,
    "name": "fakeimage123456",
    "progress": 100,
    "status": "ACTIVE",
    "updated": "2011-01-01T01:02:03Z"
},
{
    "OS-EXT-IMG-SIZE:size": "74185822",
    "created": "2011-01-01T01:02:03Z",
    "id": "76fa36fc-c930-4bf3-8c8a-ea2a2420deb6",
    "links": [
        {
            "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/76fa36fc-c930-4bf3-8c8a-ea2a2420deb6",
            "rel": "self"
        },
        {

```

8c8a-ea2a2420deb6", "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/76fa36fc-c930-4bf3-
"rel": "bookmark"


```

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

```

GET

/images/{image_id}

Show Image Details

Shows details for an image.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|----------|------|--------|------------------------|
| image_id | path | string | The UUID of the image. |

Response

| Name | In | Type | Description |
|----------|------|---------|---------------------------------------------------------------------------------------------------|
| images | body | array | An array of Image objects. |
| id | body | string | The ID of the Image. |
| name | body | string | The display name of an Image. |
| minRam | body | integer | The minimum amount of RAM an image requires to function, in MB. For example, 512. |
| minDisk | body | integer | The minimum amount of disk space an image requires to boot, in GiB. For example, 100. |
| metadata | body | object | Metadata key and value pairs. The maximum size for each metadata key and value pair is 255 bytes. |

| Name | In | Type | Description |
|------------------------------|------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| created | body | string | <p>The date and time when the resource was created. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| updated | body | string | <p>The date and time when the resource was updated. The date and time stamp format is ISO 8601</p> <p>CCYY-MM-DDThh:mm:ss±hh:mm</p> <p>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.</p> |
| status | body | string | <p>The status of image, as a string. This can be one of:</p> <ul style="list-style-type: none"> • ACTIVE: image is in active state • SAVING: image is in queued or in saving process • DELETED: image is deleted or in progress of deletion • ERROR: image is in error state • UNKNOWN: image is in unknown state |
| progress | body | integer | <p>A percentage value of the image save progress. This can be one of:</p> <ul style="list-style-type: none"> • ACTIVE: 100 • SAVING: 25 or 50 |
| links | body | array | <p>Links to the resources in question. See API Guide / Links and References for more info.</p> |
| server (Optional) | body | object | <p>The server booted from image.</p> |
| OS-EXT-IMG-SIZE:size | body | integer | <p>The size of the image.</p> |
| OS-DCF:diskConfig (Optional) | body | string | <p>Controls how the API partitions the disk when you create, rebuild, or resize servers. A server inherits the <code>OS-DCF:diskConfig</code> value from the image from which it was created, and an image inherits the <code>OS-DCF:diskConfig</code> 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</p> |

| Name | In | Type | Description |
|------|----|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | <p>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:</p> <ul style="list-style-type: none"> 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. |

Example Show Image Details: JSON response

```
{
  "images": [
    {
      "OS-DCF:diskConfig": "AUTO",
      "OS-EXT-IMG-SIZE:size": "74185822",
      "created": "2011-01-01T01:02:03Z",
      "id": "70a599e0-31e7-49b7-b260-868f441e862b",
      "links": [
        {
          "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/70a599e0-31e7-49b7-b260-868f441e862b",
          "rel": "self"
        },
        {
          "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/70a599e0-31e7-49b7-b260-868f441e862b",
          "rel": "bookmark"
        },
        {
          "href": "http://glance.openstack.example.com/images/70a599e0-31e7-49b7-b260-868f441e862b",
          "rel": "alternate",
          "type": "application/vnd.openstack.image"
        }
      ],
      "metadata": {
        "architecture": "x86_64",
        "auto_disk_config": "True",
        "kernel_id": "nokernel",
        "ramdisk_id": "nokernel"
      },
      "minDisk": 0,
      "minRam": 0,
      "name": "fakeimage7",
      "progress": 100,
      "status": "ACTIVE",
      "updated": "2011-01-01T01:02:03Z"
    },
    {
      "OS-EXT-IMG-SIZE:size": "74185822",
      "created": "2011-01-01T01:02:03Z",
      "id": "155d900f-4e14-4e4c-a73d-069cbf4541e6",
      "links": [
        {
          "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/155d900f-4e14-4e4c-a73d-069cbf4541e6",
          "rel": "self"
        },
        {
          "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/155d900f-4e14-4e4c-a73d-069cbf4541e6",
          "rel": "bookmark"
        },
        {
          "href": "http://glance.openstack.example.com/images/155d900f-4e14-4e4c-a73d-069cbf4541e6",
          "rel": "alternate",
          "type": "application/vnd.openstack.image"
        }
      ],
      "metadata": {
        "architecture": "x86_64",
        "kernel_id": "nokernel",
        "ramdisk_id": "nokernel"
      },
      "minDisk": 0,

```

```

        "minRam": 0,
        "name": "fakeimage123456",
        "progress": 100,
        "status": "ACTIVE",
        "updated": "2011-01-01T01:02:03Z"
    },
    {
        "OS-EXT-IMG-SIZE:size": "74185822",
        "created": "2011-01-01T01:02:03Z",
        "id": "a2459075-d96c-40d5-893e-577ff92e721c",
        "links": [
            {
                "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/a2459075-d96c-40d5-893e-577ff92e721c",
                "rel": "self"
            },
            {
                "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/a2459075-d96c-40d5-893e-577ff92e721c",
                "rel": "bookmark"
            },
            {
                "href": "http://glance.openstack.example.com/images/a2459075-d96c-40d5-893e-577ff92e721c",
                "rel": "alternate",
                "type": "application/vnd.openstack.image"
            }
        ],
        "metadata": {
            "kernel_id": "nokernel",
            "ramdisk_id": "nokernel"
        },
        "minDisk": 0,
        "minRam": 0,
        "name": "fakeimage123456",
        "progress": 100,
        "status": "ACTIVE",
        "updated": "2011-01-01T01:02:03Z"
    },
    {
        "OS-DCF:diskConfig": "MANUAL",
        "OS-EXT-IMG-SIZE:size": "74185822",
        "created": "2011-01-01T01:02:03Z",
        "id": "a440c04b-79fa-479c-bed1-0b816eaec379",
        "links": [
            {
                "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/a440c04b-79fa-479c-bed1-0b816eaec379",
                "rel": "self"
            },
            {
                "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/a440c04b-79fa-479c-bed1-0b816eaec379",
                "rel": "bookmark"
            },
            {
                "href": "http://glance.openstack.example.com/images/a440c04b-79fa-479c-bed1-0b816eaec379",
                "rel": "alternate",
                "type": "application/vnd.openstack.image"
            }
        ],
        "metadata": {
            "architecture": "x86_64",
            "auto_disk_config": "False",
            "kernel_id": "nokernel",
            "ramdisk_id": "nokernel"
        },
        "minDisk": 0,
        "minRam": 0,
        "name": "fakeimage6",
        "progress": 100,
        "status": "ACTIVE",
        "updated": "2011-01-01T01:02:03Z"
    },
    {
        "OS-EXT-IMG-SIZE:size": "74185822",
        "created": "2011-01-01T01:02:03Z",
        "id": "c905cedb-7281-47e4-8a62-f26bc5fc4c77",
        "links": [
            {
                "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/c905cedb-7281-47e4-8a62-f26bc5fc4c77",
                "rel": "self"
            },
            {
                "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/c905cedb-7281-47e4-8a62-f26bc5fc4c77",
                "rel": "bookmark"
            },
            {
                "href": "http://glance.openstack.example.com/images/c905cedb-7281-47e4-8a62-f26bc5fc4c77",
                "rel": "alternate",
                "type": "application/vnd.openstack.image"
            }
        ],
        "metadata": {
            "kernel_id": "155d900f-4e14-4e4c-a73d-069cbf4541e6",
            "ramdisk_id": null
        },
    },

```

```

        "minDisk": 0,
        "minRam": 0,
        "name": "fakeimage123456",
        "progress": 100,
        "status": "ACTIVE",
        "updated": "2011-01-01T01:02:03Z"
    },
    {
        "OS-EXT-IMG-SIZE:size": "74185822",
        "created": "2011-01-01T01:02:03Z",
        "id": "cedef40a-ed67-4d10-800e-17455edce175",
        "links": [
            {
                "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/cedef40a-ed67-4d10-800e-17455edce175",
                "rel": "self"
            },
            {
                "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/cedef40a-ed67-4d10-800e-17455edce175",
                "rel": "bookmark"
            },
            {
                "href": "http://glance.openstack.example.com/images/cedef40a-ed67-4d10-800e-17455edce175",
                "rel": "alternate",
                "type": "application/vnd.openstack.image"
            }
        ],
        "metadata": {
            "kernel_id": "nokernel",
            "ramdisk_id": "nokernel"
        },
        "minDisk": 0,
        "minRam": 0,
        "name": "fakeimage123456",
        "progress": 100,
        "status": "ACTIVE",
        "updated": "2011-01-01T01:02:03Z"
    },
    {
        "OS-EXT-IMG-SIZE:size": "74185822",
        "created": "2011-01-01T01:02:03Z",
        "id": "76fa36fc-c930-4bf3-8c8a-ea2a2420deb6",
        "links": [
            {
                "href": "http://openstack.example.com/v2/6f70656e737461636b20342065766572/images/76fa36fc-c930-4bf3-8c8a-ea2a2420deb6",
                "rel": "self"
            },
            {
                "href": "http://openstack.example.com/6f70656e737461636b20342065766572/images/76fa36fc-c930-4bf3-8c8a-ea2a2420deb6",
                "rel": "bookmark"
            },
            {
                "href": "http://glance.openstack.example.com/images/76fa36fc-c930-4bf3-8c8a-ea2a2420deb6",
                "rel": "alternate",
                "type": "application/vnd.openstack.image"
            }
        ],
        "metadata": {
            "kernel_id": "nokernel",
            "ramdisk_id": "nokernel"
        },
        "minDisk": 0,
        "minRam": 0,
        "name": "fakeimage123456",
        "progress": 100,
        "status": "ACTIVE",
        "updated": "2011-01-01T01:02:03Z"
    }
]
}

```

DELETE

/images/{image_id}

Delete Image

Deletes an image.

Normal response codes: 204

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

Request

| Name | In | Type | Description |
|----------|------|--------|------------------------|
| image_id | path | string | The UUID of the image. |

Response

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

GET

/images/{image_id}/metadata

List Image Metadata

List metadata of an image.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|----------|------|--------|------------------------|
| image_id | path | string | The UUID of the image. |

Response

| Name | In | Type | Description |
|----------|------|--------|---------------------------------------------------------------------------------------------------|
| metadata | body | object | Metadata key and value pairs. The maximum size for each metadata key and value pair is 255 bytes. |

Example List Image Metadata Details: JSON response

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

POST

/images/{image_id}/metadata

Create Image Metadata

Create an image metadata.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|----------|------|--------|---------------------------------------------------------------------------------------------------|
| image_id | path | string | The UUID of the image. |
| metadata | body | object | Metadata key and value pairs. The maximum size for each metadata key and value pair is 255 bytes. |

Example Create Image Metadata: JSON request

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

Response

| Name | In | Type | Description |
|----------|------|--------|---------------------------------------------------------------------------------------------------|
| metadata | body | object | Metadata key and value pairs. The maximum size for each metadata key and value pair is 255 bytes. |

Example Create Image Metadata: JSON response

```
{
  "metadata": {
    "Label": "UpdatedImage",
    "architecture": "x86_64",
    "auto_disk_config": "True",
    "kernel_id": "False",
    "ramdisk_id": "nokernel"
  }
}
```

PUT

/images/{image_id}/metadata

Update Image Metadata

Update an image metadata

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|----------|------|--------|-------------------------------------------|
| image_id | path | string | The UUID of the image. |
| metadata | body | object | Metadata key and value pairs. The maximum |

| Name | In | Type | Description |
|------|----|------|---------------------------------------------------------|
| | | | size for each metadata key and value pair is 255 bytes. |

Example Update Image Metadata: JSON request

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

Response

| Name | In | Type | Description |
|----------|------|--------|---------------------------------------------------------------------------------------------------|
| metadata | body | object | Metadata key and value pairs. The maximum size for each metadata key and value pair is 255 bytes. |

Example Update Image Metadata: JSON response

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

GET

/images/{image_id}/metadata/{key}

Show Image Metadata Item

Shows metadata item, by key, for an image.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|----------|------|--------|-----------------------------------------------------------------------|
| image_id | path | string | The UUID of the image. |
| key | path | string | The metadata item key, as a string. Maximum length is 255 characters. |

Response

| Name | In | Type | Description |
|------|------|--------|--------------------------------------------|
| meta | body | object | The object of detailed key metadata items. |

Example Show Image Metadata Item Details: JSON response

```
{
  "meta": {
    "kernel_id": "nokernel"
  }
}
```

PUT

/images/{image_id}/metadata/{key}

Create Or Update Image Metadata Item

Creates or updates a metadata item, by key, for an image.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|----------|------|--------|-----------------------------------------------------------------------|
| image_id | path | string | The UUID of the image. |
| key | path | string | The metadata item key, as a string. Maximum length is 255 characters. |
| meta | body | object | The object of detailed key metadata items. |

Example Create Or Update Image Metadata Item: JSON request

```
{
  "meta": {
    "auto_disk_config": "False"
  }
}
```

Response

| Name | In | Type | Description |
|------|------|--------|--------------------------------------------|
| meta | body | object | The object of detailed key metadata items. |

Example Create Or Update Image Metadata Item: JSON response

```
{
  "meta": {
    "auto_disk_config": "False"
  }
}
```

DELETE

/images/{image_id}/metadata/{key}

Delete Image Metadata Item

Deletes a metadata item, by key, for an image.

Normal response codes: 204

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

Request

| Name | In | Type | Description |
|----------|------|--------|-----------------------------------------------------------------------|
| image_id | path | string | The UUID of the image. |
| key | path | string | The metadata item key, as a string. Maximum length is 255 characters. |

Response

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

Bare metal nodes (os-baremetal-nodes) (DEPRECATED)

Warning

These APIs are proxy calls to the Ironi service. They exist for legacy compatibility, but no new applications should use them. Nova has deprecated all the proxy APIs and users should use the native APIs instead. These will fail with a 404 after microversion 2.35. See: [Relevant Bare metal APIs](#).

Bare metal nodes.

GET

/os-baremetal-nodes

List Bare Metal Nodes

Lists the bare metal nodes known by the compute environment.

Normal response codes: 202

Error response codes: unauthorized(401), forbidden(403), notImplemented(501)

Response

| Name | In | Type | Description |
|------------|------|--------|-----------------------------------------------------------------------------------------------------------------------------------------|
| nodes | body | array | An array of baremetal node objects. |
| id | body | string | UUID of the baremetal node. |
| interfaces | body | array | A list of interface objects for active interfaces on the baremetal node. Each will have an <code>address</code> field with the address. |
| host | body | string | This will always have the value <code>IRONIC MANAGED</code> . |
| task_state | body | string | The Ironic task state for the node. See Ironic project for more details. |
| cpus | body | string | Number of CPUs the node has. Note This is a JSON string, even though it will look like an int value. |
| memory_mb | body | string | Amount of memory in MiB the node has. Note This is a JSON string, even though it will look like an int value. |
| disk_gb | body | string | Amount of disk in GiB the node has. Note This is a JSON string, even though it will look like an int value. |

Example List Bare Metal Nodes

```
{
  "nodes": [
    {
      "cpus": "2",
      "disk_gb": "10",
      "host": "IRONIC MANAGED",
      "id": "058d27fa-241b-445a-a386-08c04f96db43",
      "interfaces": [],
      "memory_mb": "1024",
      "task_state": "active"
    },
    {
      "cpus": "2",
      "disk_gb": "10",
      "host": "IRONIC MANAGED",
      "id": "e2025409-f3ce-4d6a-9788-c565cf3b1b1c",

```

```
        "interfaces": [],
        "memory_mb": "1024",
        "task_state": "active"
    }
]
```

GET

/os-baremetal-nodes/{node_id}

Show Bare Metal Node Details

Shows details for a bare metal node.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|---------|------|--------|--------------|
| node_id | path | string | The node ID. |

Response

| Name | In | Type | Description |
|---------------|------|--------|-----------------------------------------------------------------------------------------------------------------------------------------|
| node | body | object | A baremetal node object. |
| id | body | string | UUID of the baremetal node. |
| instance_uuid | body | string | UUID of the server instance on this node. |
| interfaces | body | array | A list of interface objects for active interfaces on the baremetal node. Each will have an <code>address</code> field with the address. |
| host | body | string | This will always have the value IRONIC MANAGED . |
| task_state | body | string | The Ironic task state for the node. See Ironic project for more details. |
| cpus | body | string | Number of CPUs the node has. Note This is a JSON string, even though it will look like an int value. |
| memory_mb | body | string | Amount of memory in MiB the node has. Note |

| Name | In | Type | Description |
|---------|------|--------|----------------------------------------------------------------------------------------------------------------------------------|
| | | | This is a JSON string, even though it will look like an int value. |
| disk_gb | body | string | Amount of disk in GiB the node has. Note This is a JSON string, even though it will look like an int value. |

Example Show Bare Metal Node Details

```
{
  "node": {
    "cpus": "2",
    "disk_gb": "10",
    "host": "IRONIC MANAGED",
    "id": "058d27fa-241b-445a-a386-08c04f96db43",
    "instance_uuid": "1ea4e53e-149a-4f02-9515-590c9fb2315a",
    "interfaces": [],
    "memory_mb": "1024",
    "task_state": "active"
  }
}
```

Project networks (os-tenant-networks) (DEPRECATED)

Warning

These APIs are proxy calls to the Network service. Nova has deprecated all the proxy APIs and users should use the native APIs instead. These will fail with a 404 after microversion 2.35. See: [Relevant Network APIs](#).

Creates, lists, shows information for, and deletes project networks.

GET

/os-tenant-networks

List Project Networks

Lists all project networks.

Policy defaults enable only users with the administrative role or the owner of the network 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

Example List Project Networks: JSON response

```
{
  "networks": [
    {
      "cidr": "10.0.0.0/29",
      "id": "616fb98f-46ca-475e-917e-2563e5a8cd19",
      "label": "test_0"
    },
    {
      "cidr": "10.0.0.8/29",
      "id": "616fb98f-46ca-475e-917e-2563e5a8cd20",
      "label": "test_1"
    }
  ]
}
```

POST

/os-tenant-networks

Create Project Network

Creates a project network.

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)`, `conflict(409)`, `serviceUnavailable(503)`

Example Create Project Network: JSON request

```
{
  "network": {
    "label": "public",
    "cidr": "172.0.0.0/24",
    "vlan_start": 1,
    "num_networks": 1,
    "network_size": 255
  }
}
```

Response

Example Create Project Network: JSON response

```
{
  "network": {
    "cidr": "172.0.0.0/24",
    "id": "5bbcc3c4-1da2-4437-a48a-66f15b1b13f9",
    "label": "public"
  }
}
```

GET

/os-tenant-networks/{network_id}

Show Project Network Details

Shows details for a project network.

Policy defaults enable only users with the administrative role or the owner of the network 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), itemNotFound(404)

Request

| Name | In | Type | Description |
|------------|------|--------|--------------------------|
| network_id | path | string | The UUID of the network. |

Response

Example Show Project Network Details: JSON response

```
{
  "network": {
    "cidr": "172.0.0.0/24",
    "id": "5bbcc3c4-1da2-4437-a48a-66f15b1b13f9",
    "label": "public"
  }
}
```

DELETE

/os-tenant-networks/{network_id}

Delete Project Network

Deletes a project network.

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

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|------------|------|--------|--------------------------|
| network_id | path | string | The UUID of the network. |

Response

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

Fixed IPs (os-fixed-ips) (DEPRECATED)

Warning

These APIs are proxy calls to the Network service. Nova has deprecated all the proxy APIs and users should use the native APIs instead. These will fail with a 404 after microversion 2.35.

Shows data for a fixed IP, such as host name, CIDR, and address. Also, reserves and releases a fixed IP address.

GET

/os-fixed-ips/{fixed_ip}

Show Fixed Ip Details

Shows details for a fixed IP address.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|----------|------|--------|----------------------------------|
| fixed_ip | path | string | The fixed IP of interest to you. |

Response

| Name | In | Type | Description |
|----------|------|--------|-------------------------------------------------------------------------------------------------|
| fixed_ip | body | object | A fixed IP address object. |
| address | body | string | The IP address. |
| cidr | body | string | The CIDR for address range. |
| host | body | string | The hostname of the host that manages the server that is associated with this fixed IP address. |
| hostname | body | string | The hostname of the server that is associated with this fixed IP address. |

Example Show Fixed Ip Details: JSON response

```
{
  "fixed_ip": {
    "address": "192.168.1.1",
    "cidr": "192.168.1.0/24",
    "host": "host",
    "hostname": "compute.host.pvt"
  }
}
```



```
}
```

POST

/os-fixed-ips/{fixed_ip}/action

Reserve Or Release A Fixed Ip

Reserves or releases a fixed IP.

To reserve a fixed IP address, specify `reserve` in the request body. To release a fixed IP address, specify `unreserve` in the request body.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|----------------------|------|--------|--------------------------------------------------------------------|
| fixed_ip | path | string | The fixed IP of interest to you. |
| reserve (Optional) | body | string | The attribute to reserve an IP with a value of <code>null</code> . |
| unreserve (Optional) | body | string | The attribute to release an IP with a value of <code>null</code> . |

Example Reserve Or Release A Fixed Ip: JSON request

```
{
  "reserve": null
}
```

Response

There is no body content for the response of a successful POST operation.

Floating IP DNS records (os-floating-ip-dns) (DEPRECATED)

Warning

These APIs are proxy calls to the Network service. Nova has deprecated all the proxy APIs and users should use the native APIs instead. These will fail with a 404 after microversion 2.35. See: [Relevant Network APIs](#).

Manages DNS records associated with floating IP addresses. The API dispatches requests to a DNS driver that is selected at startup.

GET

/os-floating-ip-dns

List Dns Domains

Lists registered DNS domains published by the DNS drivers.

Normal response codes: 200

Error response codes: unauthorized(401), forbidden(403), notImplemented(501)

Response

Example List Dns Domains: JSON response

```
{
  "domain_entries": [
    {
      "availability_zone": null,
      "domain": "domain1.example.org",
      "project": "project1",
      "scope": "public"
    }
  ]
}
```

PUT

/os-floating-ip-dns/{domain}

Create Or Update Dns Domain

Creates or updates a DNS domain.

Normal response codes: 200

Error response codes: badRequest(400), unauthorized(401), forbidden(403), notImplemented(501)

Request

| Name | In | Type | Description |
|--------|------|--------|---------------------------------------------------------|
| domain | path | string | The registered DNS domain that the DNS drivers publish. |

Example Create Or Update Dns Domain: JSON request

```
{
  "domain_entry": {
    "scope": "public",
    "project": "project1"
  }
}
```

Response

Example Create Or Update Dns Domain: JSON response

```
{
  "domain_entry": {
    "availability_zone": null,
```

```
    "domain": "domain1.example.org",
    "project": "project1",
    "scope": "public"
  }
}
```

DELETE

/os-floating-ip-dns/{domain}

Delete Dns Domain

Deletes a DNS domain and all associated host entries.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|--------|------|--------|---------------------------------------------------------|
| domain | path | string | The registered DNS domain that the DNS drivers publish. |

Response

GET

/os-floating-ip-dns/{domain}/entries/{ip}

List Dns Entries

Lists DNS entries for a domain and IP.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------|------|--------|---------------------------------------------------------|
| domain | path | string | The registered DNS domain that the DNS drivers publish. |
| ip | path | string | The IP address. |

Response

Example List Dns Entries: JSON response

```
{
  "dns_entries": [
    {
      "domain": "domain1.example.org",
      "id": null,
      "ip": "192.168.1.1",
      "name": "instance1",

```

```
        "type": null
    }
  ]
}
```

GET

/os-floating-ip-dns/{domain}/entries/{name}

Find Unique Dns Entry

Finds a unique DNS entry for a domain and name.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------|------|--------|---------------------------------------------------------|
| domain | path | string | The registered DNS domain that the DNS drivers publish. |
| name | body | string | The security group name. |

Response

Example Find Unique Dns Entry: JSON response

```
{
  "dns_entry": {
    "domain": "domain1.example.org",
    "id": null,
    "ip": "192.168.1.1",
    "name": "instance1",
    "type": null
  }
}
```

PUT

/os-floating-ip-dns/{domain}/entries/{name}

Create Or Update Dns Entry

Creates or updates a DNS entry.

Normal response codes: 200

Error response codes: unauthorized(401), forbidden(403), notImplemented(501)

Request

| Name | In | Type | Description |
|--------|------|--------|---------------------------------------------------------|
| domain | path | string | The registered DNS domain that the DNS drivers publish. |

| Name | In | Type | Description |
|------|------|--------|--------------------------|
| name | body | string | The security group name. |

Example Create Or Update Dns Entry: JSON request

```
{
  "dns_entry": {
    "ip": "192.168.53.11",
    "dns_type": "A"
  }
}
```

Response

Example Create Or Update Dns Entry: JSON response

```
{
  "dns_entry": {
    "domain": "domain1.example.org",
    "id": null,
    "ip": "192.168.1.1",
    "name": "instance1",
    "type": "A"
  }
}
```

DELETE

/os-floating-ip-dns/{domain}/entries/{name}

Delete Dns Entry

Deletes a DNS entry.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|--------|------|--------|---------------------------------------------------------|
| domain | path | string | The registered DNS domain that the DNS drivers publish. |
| name | body | string | The security group name. |

Response

Floating IP pools (os-floating-ip-pools) (DEPRECATED)

Warning

These APIs are proxy calls to the Network service. Nova has deprecated all the proxy APIs and users should use the native APIs instead. These will fail with a 404 after microversion 2.35. See: [Relevant Network APIs](#).

Manages groups of floating IPs.

GET

/os-floating-ip-pools

List Floating Ip Pools

Lists floating IP pools.

Policy defaults enable only users with the administrative role or user who is authorized to operate on tenant <tenant_id> 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 |
|-------------------|------|--------|-----------------------------------|
| floating_ip_pools | body | array | The floating_ip_pools object. |
| name | body | string | The name of the floating ip pool. |

Example List Floating Ip Pools: JSON response

```
{
  "floating_ip_pools": [
    {
      "name": "pool1"
    },
    {
      "name": "pool2"
    }
  ]
}
```

Floating IPs (os-floating-ips) (DEPRECATED)

Warning

These APIs are proxy calls to the Network service. Nova has deprecated all the proxy APIs and users should use the native APIs instead. These will fail with a 404 after microversion 2.35. See: [Relevant Network APIs](#).

Lists floating IP addresses for a project. Also, creates (allocates) a floating IP address for a project, shows floating IP address details, and deletes (deallocates) a floating IP address from a project.

The cloud administrator configures a pool of floating IP addresses in OpenStack Compute. The project quota defines the maximum number of floating IP addresses that you can allocate to the project. After you [allocate a floating IP address](#) for a project, you can:

- [Add \(associate\) the floating IP address](#) with an instance in the project. You can associate only one floating IP address with an instance at a time.
- [Remove \(disassociate\) the floating IP address](#) from an instance in the project.
- Delete, or deallocate, a floating IP from the project, which automatically deletes any associations for that IP address.

GET

/os-floating-ips

List Floating Ip Addresses

Lists floating IP addresses associated with the tenant or account.

Policy defaults enable only users with the administrative role or the owner of the server 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

Example List Floating Ip Addresses: JSON response

```
{
  "floating_ips": [
    {
      "fixed_ip": null,
      "id": 1,
      "instance_id": null,
      "ip": "10.10.10.1",
      "pool": "nova"
    },
    {
      "fixed_ip": null,
      "id": 2,
      "instance_id": null,
      "ip": "10.10.10.2",
      "pool": "nova"
    }
  ]
}
```

POST

/os-floating-ips

Create (Allocate) Floating Ip Address

Creates, or allocates, a floating IP address for the current project. By default, the floating IP address is allocated from the public pool.

If more than one floating IP address pool is available, use the `pool` parameter to specify from which pool to allocate the IP address.

Policy defaults enable only users with the administrative role or the owner of the server 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)`, `itemNotFound(404)`

Request

| Name | In | Type | Description |
|-----------------|------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| pool (Optional) | body | string | Pool from which to allocate the IP address. If you omit this parameter, the call allocates the floating IP address from the public pool. If no floating IP addresses are available, the call returns the 400 response code with an informational message. Policy defaults enable only users with the administrative role or the owner of the server to perform this operation. Cloud providers can change these permissions through the <code>policy.json</code> file. |

Example Create (Allocate) Floating Ip Address: JSON request

```
{
  "pool": "nova"
}
```

Response

Example Create (Allocate) Floating Ip Address: JSON response

```
{
  "floating_ip": {
    "fixed_ip": null,
    "id": 1,
    "instance_id": null,
    "ip": "10.10.10.1",
    "pool": "nova"
  }
}
```

GET

/os-floating-ips/{floating_ip_id}

Show Floating Ip Address Details

Shows details for a floating IP address, by ID, that is associated with the tenant or account.

Policy defaults enable only users with the administrative role or the owner of the server 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)`, `itemNotFound(404)`

Request

| Name | In | Type | Description |
|-----------------------------|------|--------|------------------------------------|
| <code>floating_ip_id</code> | path | string | The ID of the floating IP address. |

Response

Example Show Floating Ip Address Details: JSON response

```
{
  "floating_ip": {
    "fixed_ip": null,
    "id": 1,
    "instance_id": null,
    "ip": "10.10.10.1",
    "pool": "nova"
  }
}
```

DELETE

`/os-floating-ips/{floating_ip_id}`

Delete (Deallocate) Floating Ip Address

Deletes, or deallocates, a floating IP address from the current project and returns it to the pool from which it was allocated.

If the IP address is still associated with a running instance, it is automatically disassociated from that instance.

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

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|-----------------------------|------|--------|------------------------------------|
| <code>floating_ip_id</code> | path | string | The ID of the floating IP address. |

Response

Floating IPs bulk (os-floating-ips-bulk) (DEPRECATED)

Warning

These APIs are proxy calls to the Network service. Nova has deprecated all the proxy APIs and users should use the native APIs instead. These will fail with a 404 after microversion 2.35. See: [Relevant Network APIs](#).

(nova-network only) Bulk-creates, deletes, and lists floating IPs. Default pool name is nova.

To view available pools, use the `os-floating-ip-pools` extension.

GET

/os-floating-ips-bulk

List Floating Ips

Lists all floating IPs.

Normal response codes: 200

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

Response

| Name | In | Type | Description |
|------------------|------|--------|--------------------------------------------------------------|
| floating_ip_info | body | array | An array of floating ip objects. |
| address | body | string | The floating ip address. |
| fixed_ip | body | string | Fixed IP associated with floating IP network. |
| instance_uuid | body | string | The UUID of the server. |
| interface | body | string | Virtual interface for the floating ip address. |
| pool | body | string | The name of the floating ip pool. |
| project_id | body | string | The project id under which the bulk ip addresses are created |

Example List Floating Ips: JSON response

```
{
  "floating_ip_info": [
    {
```

```

        "address": "10.10.10.1",
        "instance_uuid": null,
        "fixed_ip": null,
        "interface": "eth0",
        "pool": "nova",
        "project_id": null
    },
    {
        "address": "10.10.10.2",
        "instance_uuid": null,
        "fixed_ip": null,
        "interface": "eth0",
        "pool": "nova",
        "project_id": null
    },
    {
        "address": "10.10.10.3",
        "instance_uuid": null,
        "fixed_ip": null,
        "interface": "eth0",
        "pool": "nova",
        "project_id": null
    }
]
}

```

POST

/os-floating-ips-bulk

Create Floating Ips

Bulk-creates floating IPs.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|--------------------------|------|--------|-------------------------------------------------------------|
| floating_ips_bulk_create | body | object | The floating ip bulk address object. |
| ip_range | body | string | The range of IP addresses to use for creating floating IPs. |
| interface (Optional) | body | string | Virtual interface for the floating ip address |
| pool (Optional) | body | string | The name of the floating ip pool |

Example Create Floating Ips: JSON request

```

{
  "floating_ips_bulk_create": {
    "ip_range": "192.168.1.0/24",
    "pool": "nova",
    "interface": "eth0"
  }
}

```

Response

| Name | In | Type | Description |
|--------------------------|------|--------|-------------------------------------------------------------|
| floating_ips_bulk_create | body | object | The floating ip bulk address object. |
| interface | body | string | Virtual interface for the floating ip address. |
| ip_range | body | string | The range of IP addresses to use for creating floating IPs. |
| pool | body | string | The name of the floating ip pool. |

Example Create Floating Ips: JSON response

```
{
  "floating_ips_bulk_create": {
    "interface": "eth0",
    "ip_range": "192.168.1.0/24",
    "pool": "nova"
  }
}
```

PUT

/os-floating-ips-bulk/delete

Bulk-Delete Floating Ips

Bulk-deletes floating IPs.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|----------|------|--------|-------------------------------------------------------------------|
| ip_range | body | string | The range of IP addresses from which to bulk-delete floating IPs. |

Example Bulk-Delete Floating Ips: JSON request

```
{
  "ip_range": "192.168.1.0/24"
}
```

Response

| Name | In | Type | Description |
|--------------------------|------|--------|-------------------------------------------------------------------|
| floating_ips_bulk_delete | body | string | The range of IP addresses from which to bulk-delete floating IPs. |

Example Bulk-Delete Floating Ips: JSON response

```
{
  "floating_ips_bulk_delete": "192.168.1.0/24"
}
```

GET

/os-floating-ips-bulk/{host_name}

List Floating Ips By Host

Lists all floating IPs for a host.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-----------|------|--------|-----------------------|
| host_name | path | string | The name of the host. |

Response

| Name | In | Type | Description |
|------------------|------|--------|--------------------------------------------------------------|
| floating_ip_info | body | array | An array of floating ip objects. |
| address | body | string | The floating ip address. |
| fixed_ip | body | string | Fixed IP associated with floating IP network. |
| instance_uuid | body | string | The UUID of the server. |
| interface | body | string | Virtual interface for the floating ip address. |
| pool | body | string | The name of the floating ip pool. |
| project_id | body | string | The project id under which the bulk ip addresses are created |

Example List Floating Ips By Host: JSON response

```
{
  "floating_ip_info": [
    {
      "address": "10.10.10.3",
      "instance_uuid": null,
      "fixed_ip": null,
      "interface": "eth0",
      "pool": "nova",
      "project_id": null
    }
  ]
}
```

Ping instances (os-fping) (DEPRECATED)

Warning

This API only works with `nova-network` which is deprecated. It should be avoided in any new applications. These will fail with a 404 after microversion 2.35.

Pings instances and reports which instances are alive.

GET

/os-fping

Ping Instances

Runs the fping utility to ping instances and reports which instances are alive.

Specify the `all_tenants=1` query parameter to ping instances for all tenants. For example:

```
GET /os-fping?all_tenants=1
```

Specify the `include` and `exclude` query parameters to filter the results. For example:

```
GET /os-fping?all_tenants=1&include=uuid1,uuid2&exclude=uuid3,uuid4
```

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

Normal response codes: 200

Error response codes: `serviceUnavailable(503)`, `unauthorized(401)`, `forbidden(403)`

Request

| Name | In | Type | Description |
|-------------------------------------|-------|---------|-------------------------------------------------------------------------------------------------------------------------------------------|
| <code>all_tenants</code> (Optional) | query | integer | Specify the <code>all_tenants=1</code> query parameter to ping instances for all tenants. By default this is only allowed by admin users. |
| <code>include</code> (Optional) | query | string | Specify <code>include=uuid[,uuid...]</code> to include the instances in the results. |
| <code>exclude</code> (Optional) | query | string | Specify <code>exclude=uuid[,uuid...]</code> to exclude the instances from the results. |

Response

| Name | In | Type | Description |
|----------------------|------|-------|----------------------------------------|
| <code>servers</code> | body | array | A list of <code>server</code> objects. |

| Name | In | Type | Description |
|-----------------------|------|---------|-------------------------------------------------------------------------------------|
| alive | body | boolean | Returns true if the instance is alive. |
| id | body | string | The UUID of the server. |
| project_id (Optional) | body | string | The UUID of the project. If omitted, the project ID defaults to the calling tenant. |

Example Ping Instances

```
{
  "servers": [
    {
      "alive": false,
      "id": "1d1aea35-472b-40cf-9337-8eb68480aaa1",
      "project_id": "6f70656e737461636b20342065766572"
    }
  ]
}
```

GET

/os-fping/{instance_id}

Ping An Instance

Runs the fping utility to ping an instance and reports whether the instance is alive.

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

Normal response codes: 200

Error response codes: serviceUnavailable(503), unauthorized(401), forbidden(403), itemNotFound(404)

Request

| Name | In | Type | Description |
|-------------|------|--------|---------------------------|
| instance_id | path | string | The UUID of the instance. |

Response

| Name | In | Type | Description |
|-----------------------|------|---------|------------------------------------------|
| server | body | object | A server object. |
| alive | body | boolean | Returns true if the instance is alive. |
| id | body | string | The UUID of the server. |
| project_id (Optional) | body | string | The UUID of the project. If omitted, the |

| Name | In | Type | Description |
|------|----|------|--------------------------------------------|
| | | | project ID defaults to the calling tenant. |

Example Ping An Instance

```
{
  "server": {
    "alive": false,
    "id": "f5e6fd6d-c0a3-4f9e-aabf-d69196b6d11a",
    "project_id": "6f70656e737461636b20342065766572"
  }
}
```

Security groups (os-security-groups) (DEPRECATED)

Warning

These APIs are proxy calls to the Network service. Nova has deprecated all the proxy APIs and users should use the native APIs instead. These will fail with a 404 after microversion 2.35. See: [Relevant Network APIs](#).

Lists, shows information for, creates, updates and deletes security groups.

GET

/os-security-groups

List Security Groups

Lists security groups.

Normal response codes: 200

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

Response

| Name | In | Type | Description |
|----------------------------|------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 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. |
| description | body | string | Security group description. |
| id | body | string | The security group name or UUID. |

| Name | In | Type | Description |
|-----------|------|--------|--------------------------------------------------|
| name | body | string | The security group name. |
| rules | body | array | A security group rules object. |
| tenant_id | body | string | The UUID of the tenant in a multi-tenancy cloud. |

Example List security groups: JSON response

```
{
  "security_groups": [
    {
      "description": "default",
      "id": 1,
      "name": "default",
      "rules": [],
      "tenant_id": "6f70656e737461636b20342065766572"
    }
  ]
}
```

POST

/os-security-groups

Create Security Group

Creates a security group.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|----------------|------|--------|--------------------------------------------------------|
| security_group | body | string | Specify the security_group action in the request body. |
| name | body | string | The security group name. |
| description | body | string | Security group description. |

Example Create security group: JSON request

```
{
  "security_group": {
    "name": "test",
    "description": "description"
  }
}
```

Response

| Name | In | Type | Description |
|----------------|------|--------|--------------------------------------------------------|
| security_group | body | string | Specify the security_group action in the request body. |
| description | body | string | Security group description. |
| id | body | string | The security group name or UUID. |
| name | body | string | The security group name. |
| rules | body | array | A security group rules object. |
| tenant_id | body | string | The UUID of the tenant in a multi-tenancy cloud. |

Example Create security group: JSON response

```
{
  "security_group": {
    "description": "default",
    "id": 1,
    "name": "default",
    "rules": [],
    "tenant_id": "6f70656e7374616336b20342065766572"
  }
}
```

GET

/os-security-groups/{security_group_id}

Show Security Group Details

Shows details for a security group.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-------------------|------|--------|-------------------------------|
| security_group_id | path | string | The ID of the security group. |

Response

| Name | In | Type | Description |
|----------------|------|--------|--------------------------------------------------------|
| security_group | body | string | Specify the security_group action in the request body. |
| description | body | string | Security group description. |

| Name | In | Type | Description |
|-----------|------|--------|--------------------------------------------------|
| id | body | string | The security group name or UUID. |
| name | body | string | The security group name. |
| rules | body | array | A security group rules object. |
| tenant_id | body | string | The UUID of the tenant in a multi-tenancy cloud. |

Example Show security group: JSON response

```
{
  "security_group": {
    "description": "default",
    "id": 1,
    "name": "default",
    "rules": [],
    "tenant_id": "6f70656e737461636b20342065766572"
  }
}
```

PUT

/os-security-groups/{security_group_id}

Update Security Group

Updates a security group.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|-------------------|------|--------|-------------------------------|
| security_group_id | path | string | The ID of the security group. |
| name | body | string | The security group name. |
| description | body | string | Security group description. |

Example Update security group: JSON request

```
{
  "security_group": {
    "name": "test",
    "description": "description"
  }
}
```

Response

| Name | In | Type | Description |
|----------------|------|--------|--------------------------------------------------------|
| security_group | body | string | Specify the security_group action in the request body. |
| description | body | string | Security group description. |
| id | body | string | The security group name or UUID. |
| name | body | string | The security group name. |
| rules | body | array | A security group rules object. |
| tenant_id | body | string | The UUID of the tenant in a multi-tenancy cloud. |

Example Update security group: JSON response

```
{
  "security_group": {
    "description": "default",
    "id": 1,
    "name": "default",
    "rules": [],
    "tenant_id": "6f70656e7374616336b20342065766572"
  }
}
```

DELETE

/os-security-groups/{security_group_id}

Delete Security Group

Deletes a security group.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|-------------------|------|--------|-------------------------------|
| security_group_id | path | string | The ID of the security group. |

Response

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

Rules for default security group (os-security-group-default-rules) (DEPRECATED)

Warning

This API only available with `nova-network` which is deprecated. It should be avoided in any new applications. These will fail with a 404 after microversion 2.35.

Lists, shows information for, and creates default security group rules.

GET

/os-security-group-default-rules

List Default Security Group Rules

Lists default security group rules.

Normal response codes: 200

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

Response

| Name | In | Type | Description |
|-------------|------|---------|-------------------------------------------------------------|
| from_port | body | integer | The port at start of range. |
| id | body | string | The security group name or UUID. |
| ip_protocol | body | string | The IP protocol. A valid value is ICMP, TCP, or UDP. |
| ip_range | body | string | The range of IP addresses to use for creating floating IPs. |
| cidr | body | string | The CIDR for address range. |
| to_port | body | integer | The port at end of range. |

Example List default security group rules: JSON response

```
{
  "security_group_default_rules": [
    {
      "from_port": 80,
      "id": 1,
      "ip_protocol": "TCP",
      "ip_range": {
        "cidr": "10.10.10.0/24"
      },
      "to_port": 80
    }
  ]
}
```

}

GET

/os-security-group-default-rules/{security_group_default_rule_id}

Show Default Security Group Rule Details

Shows details for a security group rule.

Normal response codes: 200

Error response codes: badRequest(400), unauthorized(401), forbidden(403), itemNotFound(404), notImplemented(501)

Request

| Name | In | Type | Description |
|--------------------------------|------|--------|--------------------------------------|
| security_group_default_rule_id | path | string | The UUID of the security group rule. |

Response

| Name | In | Type | Description |
|-------------|------|---------|-------------------------------------------------------------|
| from_port | body | integer | The port at start of range. |
| id | body | string | The security group name or UUID. |
| ip_protocol | body | string | The IP protocol. A valid value is ICMP, TCP, or UDP. |
| ip_range | body | string | The range of IP addresses to use for creating floating IPs. |
| cidr | body | string | The CIDR for address range. |
| to_port | body | integer | The port at end of range. |

Example Show default security group rule: JSON response

```
{
  "security_group_default_rule": {
    "from_port": 80,
    "id": 1,
    "ip_protocol": "TCP",
    "ip_range": {
      "cidr": "10.10.10.0/24"
    },
    "to_port": 80
  }
}
```

POST

/os-security-group-default-rules

Create Default Security Group Rule

Creates a default security group rule.

If you specify a source port (`from_port`) or destination port (`to_port`) value, you must specify an IP protocol (`ip_protocol`) value. Otherwise, the operation returns the `Bad Request (400)` response code.

Normal response codes: 200

Error response codes: `badRequest(400)`, `unauthorized(401)`, `forbidden(403)`, `conflict(409)`, `notImplemented(501)`

Request

| Name | In | Type | Description |
|-------------|------|---------|------------------------------------------------------|
| id | body | string | The security group name or UUID. |
| ip_protocol | body | string | The IP protocol. A valid value is ICMP, TCP, or UDP. |
| from_port | body | integer | The port at start of range. |
| to_port | body | integer | The port at end of range. |
| cidr | body | string | The CIDR for address range. |

Example Create default security group rule: JSON request

```
{
  "security_group_default_rule": {
    "ip_protocol": "TCP",
    "from_port": "80",
    "to_port": "80",
    "cidr": "10.10.10.0/24"
  }
}
```

Response

| Name | In | Type | Description |
|-------------|------|---------|-------------------------------------------------------------|
| from_port | body | integer | The port at start of range. |
| id | body | string | The security group name or UUID. |
| ip_protocol | body | string | The IP protocol. A valid value is ICMP, TCP, or UDP. |
| ip_range | body | string | The range of IP addresses to use for creating floating IPs. |

| Name | In | Type | Description |
|---------|------|---------|-----------------------------|
| cidr | body | string | The CIDR for address range. |
| to_port | body | integer | The port at end of range. |

Example Create default security group rule: JSON response

```
{
  "security_group_default_rule": {
    "from_port": 80,
    "id": 1,
    "ip_protocol": "TCP",
    "ip_range": {
      "cidr": "10.10.10.0/24"
    },
    "to_port": 80
  }
}
```

DELETE

/os-security-group-default-rules/{security_group_default_rule_id}

Delete Default Security Group Rule

Deletes a security group rule.

Normal response codes: 204

Error response codes: badRequest(400), unauthorized(401), forbidden(403), itemNotFound(404), notImplemented(501)

Request

| Name | In | Type | Description |
|--------------------------------|------|--------|--------------------------------------|
| security_group_default_rule_id | path | string | The UUID of the security group rule. |

Response

Rules for security group (os-security-group-rules) (DEPRECATED)

Warning

These APIs are proxy calls to the Network service. Nova has deprecated all the proxy APIs and users should use the native APIs instead. These will fail with a 404 after microversion 2.35. See: [Relevant Network APIs](#).

Creates and deletes security group rules.

POST

Create Security Group Rule

Creates a rule for a security group. Either `cidr` or `group_id` must be specified when creating a rule.

Note

`nova-network` only supports ingress rules. If you want to define egress rules you must use the Neutron networking service.

Normal response codes: 200

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

Request

| Name | In | Type | Description |
|----------------------------------|------|---------|------------------------------------------------------|
| <code>security_group_rule</code> | body | object | A <code>security_group_rule</code> object. |
| <code>parent_group_id</code> | body | string | Security group ID. |
| <code>ip_protocol</code> | body | string | The IP protocol. A valid value is ICMP, TCP, or UDP. |
| <code>from_port</code> | body | integer | The port at start of range. |
| <code>to_port</code> | body | integer | The port at end of range. |
| <code>cidr</code> (Optional) | body | string | The CIDR for address range. |
| <code>group_id</code> (Optional) | body | string | The source security group ID. |

Response

The `group` is empty if `group_id` was not provided on the request.

The `ip_range` is empty if `cidr` was not provided on the request.

| Name | In | Type | Description |
|----------------------------------|------|---------|--------------------------------------------------------------------------|
| <code>security_group_rule</code> | body | object | A <code>security_group_rule</code> object. |
| <code>ip_protocol</code> | body | string | The IP protocol. A valid value is ICMP, TCP, or UDP. |
| <code>from_port</code> | body | integer | The port at start of range. |
| <code>to_port</code> | body | integer | The port at end of range. |
| <code>ip_range</code> | body | object | An IP range object. Includes the security group rule <code>cidr</code> . |

| Name | In | Type | Description |
|-----------------|------|--------|-------------------------------------------------------------------------------------------------------|
| cidr (Optional) | body | string | The CIDR for address range. |
| id | body | string | The security group rule ID. |
| group | body | object | A group object. Includes the <code>tenant_id</code> and the source security group <code>name</code> . |
| parent_group_id | body | string | Security group ID. |
| name | body | string | The security group name. |
| tenant_id | body | string | The UUID of the tenant that owns this security group. |

DELETE

/os-security-group-rules/{security_group_rule_id}

Delete Security Group Rule

Deletes a security group rule.

Normal response codes: 202

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

Request

| Name | In | Type | Description |
|------------------------|------|--------|------------------------------------|
| security_group_rule_id | path | string | The ID of the security group rule. |

Response

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