

# OpenStack Compute

## Block Storage Extension

(Nov 6, 2011)



# OpenStack Compute Block Storage Extension

Copyright © 2011 OpenStack LLC. All rights reserved.

This document is intended for software developers interested in using the Block Storage Extension to the OpenStack Compute Application Programming Interface (API).

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

## Table of Contents

1. About This Extension .....	1
1.1. Document Change History .....	3
2. Summary of Changes .....	4
2.1. New Headers .....	4
2.2. New Faults .....	4
2.3. New Resources .....	4
2.3.1. Volumes .....	7
2.3.2. Backups .....	30
2.3.3. Types .....	51
2.4. New Actions .....	55
2.5. Changes to the OpenStack Compute API .....	55

## List of Tables

2.1. List Volumes Request Parameters .....	8
2.2. Detailed List Volumes Request Parameters .....	10
2.3. Create Volume Request Parameters .....	14
2.4. Create Volume Response Parameters .....	15
2.5. Update Volume Name Request Parameters .....	17
2.6. Get Volume Details Request Parameters .....	20
2.7. Delete Volume Request Parameters .....	22
2.8. Create or Replace Metadata Request Parameters .....	23
2.9. Update Metadata Request Parameters .....	25
2.10. List Metadata Request Parameters .....	27
2.11. Get Metadata Item Request Parameters .....	28
2.12. Create or Update Metadata Item Request Parameters .....	29
2.13. Delete Metadata Item Request Parameters .....	30
2.14. List Backups Request Parameters .....	31
2.15. Detailed List Backups Request Parameters .....	33
2.16. Create Backup Request Parameters .....	36
2.17. Create Backup Response Parameters .....	37
2.18. Update Backup Name Request Parameters .....	39
2.19. Get Backup Details Request Parameters .....	41
2.20. Delete Backup Request Parameters .....	43
2.21. Create or Replace Metadata Request Parameters .....	44
2.22. Update Metadata Request Parameters .....	46
2.23. List Metadata Request Parameters .....	48
2.24. Get Metadata Item Request Parameters .....	49
2.25. Create or Update Metadata Item Request Parameters .....	50
2.26. Delete Metadata Item Request Parameters .....	51
2.27. List Types Request Parameters .....	52
2.28. Get Type Details Request Parameters .....	54

## List of Examples

1.1. Extension Query Response: XML .....	2
1.2. Extension Query Response: JSON .....	3
2.1. List Volumes Response: XML .....	8
2.2. List Volumes Response: JSON .....	9
2.3. Detailed List Volumes Response: XML .....	10
2.4. Detailed List Volumes Response: JSON .....	11
2.5. Create Volume Request: XML .....	14
2.6. Create Volume Request: JSON .....	14
2.7. Create Volume Response: XML .....	15
2.8. Create Volume Response: JSON .....	15
2.9. Update Volume Name Request: XML .....	17
2.10. Update Volume Name Request: JSON .....	17
2.11. Update Volume Name Response: XML .....	17
2.12. Update Volume Name Response: JSON .....	18
2.13. Get Volume Details Response: XML .....	20
2.14. Get Volume Details Response: JSON .....	21
2.15. Create or Replace Metadata Request: XML .....	23
2.16. Create or Replace Metadata Request: JSON .....	23
2.17. Create or Replace Metadata Response: XML .....	23
2.18. Create or Replace Metadata Response: JSON .....	24
2.19. Update Metadata Request: XML .....	25
2.20. Update Metadata Request: JSON .....	25
2.21. Update Metadata Response: XML .....	25
2.22. Update Metadata Response: JSON .....	25
2.23. List Metadata Response: XML .....	27
2.24. List Metadata Response: JSON .....	27
2.25. Get Metadata Item Response: XML .....	28
2.26. Get Metadata Item Response: JSON .....	28
2.27. Create or Update Metadata Item Request: XML .....	29
2.28. Create or Update Metadata Item Request: JSON .....	29
2.29. Create or Update Metadata Item Response: XML .....	29
2.30. Create or Update Metadata Item Response: JSON .....	29
2.31. List Backups Response: XML .....	31
2.32. List Backups Response: JSON .....	32
2.33. Detailed List Backups Response: XML .....	33
2.34. Detailed List Backups Response: JSON .....	34
2.35. Create Backup Request: XML .....	36
2.36. Create Backup Request: JSON .....	36
2.37. Create Backup Response: XML .....	37
2.38. Create Backup Response: JSON .....	37
2.39. Update Backup Name Request: XML .....	39
2.40. Update Backup Name Request: JSON .....	39
2.41. Update Backup Name Response: XML .....	39
2.42. Update Backup Name Response: JSON .....	40
2.43. Get Backup Details Response: XML .....	41
2.44. Get Backup Details Response: JSON .....	41
2.45. Create or Replace Metadata Request: XML .....	44
2.46. Create or Replace Metadata Request: JSON .....	44

2.47. Create or Replace Metadata Response: XML .....	44
2.48. Create or Replace Metadata Response: JSON .....	45
2.49. Update Metadata Request: XML .....	46
2.50. Update Metadata Request: JSON .....	46
2.51. Update Metadata Response: XML .....	46
2.52. Update Metadata Response: JSON .....	46
2.53. List Metadata Response: XML .....	48
2.54. List Metadata Response: JSON .....	48
2.55. Get Metadata Item Response: XML .....	49
2.56. Get Metadata Item Response: JSON .....	49
2.57. Create or Update Metadata Item Request: XML .....	50
2.58. Create or Update Metadata Item Request: JSON .....	50
2.59. Create or Update Metadata Item Response: XML .....	50
2.60. Create or Update Metadata Item Response: JSON .....	50
2.61. List Types Response: XML .....	52
2.62. List Types Response: JSON .....	52
2.63. Get Type Details Response: XML .....	54
2.64. Get Type Details Response: JSON .....	54

# 1. About This Extension

Name	Block Storage Extension
Namespace	<a href="http://docs.openstack.org/servers/api/ext/volumes/v1.0">http://docs.openstack.org/servers/api/ext/volumes/v1.0</a>
Alias	OS-BS
Contact	Gabe Westmaas < <a href="mailto:gabe.westmaas@rackspace.com">gabe.westmaas@rackspace.com</a> >
Status	ALPHA
Last Update	2011-11-06
Dependencies	OpenStack Compute API v1.1 (2011-09-08)
Doc Link (PDF)	<a href="http://docs.rackspacecloud.com/servers/api/ext/os-bs.pdf">http://docs.rackspacecloud.com/servers/api/ext/os-bs.pdf</a>
Doc Link (WADL)	<a href="http://docs.rackspacecloud.com/servers/api/ext/os-bs/os-bs.wadl">http://docs.rackspacecloud.com/servers/api/ext/os-bs/os-bs.wadl</a>
Doc Link (XSD)	<a href="http://docs.rackspacecloud.com/servers/api/ext/os-bs/api.xsd">http://docs.rackspacecloud.com/servers/api/ext/os-bs/api.xsd</a>
Short Description	The extension adds block storage capabilities to the compute API.

### Example 1.1. Extension Query Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<extensions xmlns="http://docs.openstack.org/common/api/v1.0"
  xmlns:atom="http://www.w3.org/2005/Atom">
  <extension name="Block Storage Extension"
    namespace="http://docs.openstack.org/servers/api/ext/
volumes/v1.0"
    alias="OS-BS" updated="2011-11-06T00:00:00Z">
    <description>
      The extension adds block storage capabilities
      to the compute API.
    </description>
    <atom:link rel="describedby" type="application/pdf"
      href="http://docs.rackspacecloud.com/servers/api/
ext/os-bs.pdf"/>
    <atom:link rel="describedby" type="application/vnd.sun.wadl
+xml"
      href="http://docs.rackspacecloud.com/servers/api/
ext/os-bs/os-bs.wadl"/>
    <atom:link rel="describedby" type="application/xml"
      href="http://docs.rackspacecloud.com/servers/api/
ext/os-bs/api.xsd"/>
  </extension>
</extensions>
```



## Example 1.2. Extension Query Response: JSON

```
{
  "extensions": [
    {
      "name": "Block Storge Extension",
      "namespace": "http://docs.openstack.org/servers/api/ext/
volumes/v1.0",
      "alias": "OS-BS",
      "updated": "2011-11-06T00:00:00Z",
      "description": "The extension adds block storage
capabilities to the compute API.",
      "links": [
        {
          "rel": "describedby",
          "type": "application/pdf",
          "href": "http://docs.rackspacecloud.com/servers/
api/ext/os-bs.pdf"
        },
        {
          "rel": "describedby",
          "type": "application/vnd.sun.wadl+xml",
          "href": "http://docs.rackspacecloud.com/servers/
api/ext/os-bs/os-bs.wadl"
        },
        {
          "rel": "describedby",
          "type": "application/xml",
          "href": "http://docs.rackspacecloud.com/servers/
api/ext/os-bs/api.xsd"
        }
      ]
    }
  ]
}
```

## 1.1. Document Change History

The most recent changes to this document are described below.

Revision Date	Summary of Changes
Nov 6, 2011	<ul style="list-style-type: none"> <li>Initial release.</li> </ul>

## 2. Summary of Changes

The extensions adds block storage capabilities to the compute API. It introduces volumes, backups, and types as top level entities that allow the management of block storage.

### 2.1. New Headers

None.

### 2.2. New Faults

None.

### 2.3. New Resources

The following new resources are created.



Verb	URI	Description
Volume Lists		
GET	<code>/ {tenantId} /OS-BS/volumes ?changes-since=dateTime&amp; type=UUID&amp;backup=UUID&amp; name=string&amp;marker=UUID&amp; limit=int&amp;size=unsignedInt&amp; status=ExtensibleVolumeStatus</code>	List all volumes (IDs, names, links).
GET	<code>/ {tenantId} /OS-BS/volumes/ detail?changes-since=dateTime&amp; type=UUID&amp;backup=UUID&amp; name=string&amp;marker=UUID&amp; limit=int&amp;size=unsignedInt&amp; status=ExtensibleVolumeStatus</code>	List all volumes (all details).
Volume Opreations		
POST	<code>/ {tenantId} /OS-BS/volumes</code>	Creates a new volume.
PUT	<code>/ {tenantId} /OS-BS/volumes/ {id}</code>	Update the specified volume's name.
GET	<code>/ {tenantId} /OS-BS/volumes/ {id}</code>	List details of the specified volume.
DELETE	<code>/ {tenantId} /OS-BS/volumes/ {id}</code>	Delete the volume.
Volume Metadata		
PUT	<code>/ {tenantId} /OS-BS/volumes/ {id} / metadata</code>	Set metadata.
POST	<code>/ {tenantId} /OS-BS/volumes/ {id} / metadata</code>	Update metadata items.
GET	<code>/ {tenantId} /OS-BS/volumes/ {id} / metadata</code>	List metadata associated with the resource.
GET	<code>/ {tenantId} /OS-BS/volumes/ {id} / metadata/ {key}</code>	Get a metadata item associated with a resource.
PUT	<code>/ {tenantId} /OS-BS/volumes/ {id} / metadata/ {key}</code>	Set metadata item.
DELETE	<code>/ {tenantId} /OS-BS/volumes/ {id} / metadata/ {key}</code>	Delete a metadata item.
Backup Lists		
GET	<code>/ {tenantId} /OS-BS/backups?changes- since=dateTime&amp;volume=UUID&amp; name=string&amp;marker=UUID&amp; limit=int&amp;size=unsignedInt&amp; status=ExtensibleBackupStatus</code>	List all backups (IDs, names, links).
GET	<code>/ {tenantId} /OS-BS/ backups/detail?changes- since=dateTime&amp;volume=UUID&amp; name=string&amp;marker=UUID&amp; limit=int&amp;size=unsignedInt&amp; status=ExtensibleBackupStatus</code>	List all backups (all details).
Backup Opreations		
POST	<code>/ {tenantId} /OS-BS/backups</code>	Creates a new backup.
PUT	<code>/ {tenantId} /OS-BS/backups/ {id}</code>	Update the specified backup's name.
GET	<code>/ {tenantId} /OS-BS/backups/ {id}</code>	List details of the specified backup.
DELETE	<code>/ {tenantId} /OS-BS/backups/ {id}</code>	Delete the backup.
Backup Metadata		
PUT	<code>/ {tenantId} /OS-BS/backups/ {id} / metadata</code>	Set metadata.
POST	<code>/ {tenantId} /OS-BS/backups/ {id} / metadata</code>	Update metadata items.

Verb	URI	Description
GET	<code>/ {tenantId} /OS-BS/backups/{id}/metadata</code>	List metadata associated with the resource.
GET	<code>/ {tenantId} /OS-BS/backups/{id}/metadata/{key}</code>	Get a metadata item associated with a resource.
PUT	<code>/ {tenantId} /OS-BS/backups/{id}/metadata/{key}</code>	Set metadata item.
DELETE	<code>/ {tenantId} /OS-BS/backups/{id}/metadata/{key}</code>	Delete a metadata item.
Types		
GET	<code>/ {tenantId} /OS-BS/types?name=string&amp;marker=UUID&amp;limit=int</code>	List all types.
GET	<code>/ {tenantId} /OS-BS/types/{id}</code>	List details of the specified type.

## 2.3.1. Volumes

### 2.3.1.1. Volume Lists

This operations provide a list of volumes associated with a particular tenant. Volumes contain a status attribute that can be used as an indication of the current volume state. Volumes with an AVAILABLE status are available for use. A volume with an ATTACHED is currently attached to a server. Other possible values for the status attribute include: CREATING, ATTACHING, DETACHING, DELETING, DELETED, UNKNOWN, and ERROR.

The list of volumes may be filtered by type, backup, name, and status via the respective query parameters. When retrieving a list of volumes via the changes-since parameter, the list will contain volumes that have been deleted since the changes-since time (see Section 3.5, in the OpenStack Compute Dev Guide for a description of Changes-Since).

Verb	URI	Description
GET	<code>/ {tenantId} /OS-BS/volumes?changes-since=dateTime&amp;type=UUID&amp;backup=UUID&amp;name=string&amp;marker=UUID&amp;limit=int&amp;size=unsignedInt&amp;status=ExtensibleVolumeStatus</code>	List all volumes (IDs, names, links).
GET	<code>/ {tenantId} /OS-BS/volumes/detail?changes-since=dateTime&amp;type=UUID&amp;backup=UUID&amp;name=string&amp;marker=UUID&amp;limit=int&amp;size=unsignedInt&amp;status=ExtensibleVolumeStatus</code>	List all volumes (all details).

### 2.3.1.1.1. List Volumes

Verb	URI	Description
GET	/ {tenantId} /OS-BS/volumes ?changes-since=dateTime& type=UUID&backup=UUID& name=string&marker=UUID& limit=int&size=unsignedInt& status=ExtensibleVolumeStatus	List all volumes (IDs, names, links).

Normal Response Code(s): 200, 203

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413)

A list of volumes. Each volume contains IDs, names, and links – other attributes are omitted.

**Table 2.1. List Volumes Request Parameters**

Name	Style	Type	Description
changes-since	query	DateTime	The changes-since parameter is optional.
type	query	UUID	The type parameter is optional.
backup	query	UUID	The backup parameter is optional.
name	query	String	The name parameter is optional.
marker	query	UUID	The marker parameter is optional.
limit	query	Int	The limit parameter is optional.
size	query	UnsignedInt	The size parameter is optional.
status	query	ExtensibleVolumeStatus	The status parameter is optional.
tenantId	template	String	A unique identifier for the tenant who owns the resource.

### Example 2.1. List Volumes Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<volumes xmlns="http://docs.openstack.org/servers/api/ext/volumes/v1.0"
  xmlns:atom="http://www.w3.org/2005/Atom">
  <volume id="15e59938-07d5-11e1-90e3-e3dffe0c5983"
    name="Work">
    <atom:link
      rel="self"
      href="http://servers.api.openstack.org/v1.1/1234/OS-BS/volumes/
15e59938-07d5-11e1-90e3-e3dffe0c5983"/>
    <atom:link
      rel="bookmark"
      href="http://servers.api.openstack.org/1234/OS-BS/volumes/
15e59938-07d5-11e1-90e3-e3dffe0c5983"/>
    </volume>
    <volume id="4512308a-0808-11e1-bff3-83d5d70a461f"
      name="Home">
      <atom:link
        rel="self"
        href="http://servers.api.openstack.org/v1.1/1234/OS-BS/volumes/
4512308a-0808-11e1-bff3-83d5d70a461f"/>
      <atom:link
```

```

        rel="bookmark"
        href="http://servers.api.openstack.org/1234/OS-BS/volumes/
4512308a-0808-11e1-bff3-83d5d70a461f"/>
    </volume>
</volumes>

```

## Example 2.2. List Volumes Response: JSON

```

{
  "OS-BS:volumes" : [
    {
      "name" : "Work",
      "id" : "15e59938-07d5-11e1-90e3-e3dffe0c5983",
      "links" : [
        {
          "rel" : "self",
          "href" : "http://servers.api.openstack.org/v1.1/1234/OS-
BS/volumes/15e59938-07d5-11e1-90e3-e3dffe0c5983"
        },
        {
          "rel" : "bookmark",
          "href" : "http://servers.api.openstack.org/1234/OS-BS/
volumes/15e59938-07d5-11e1-90e3-e3dffe0c5983"
        }
      ]
    },
    {
      "name" : "Home",
      "id" : "4512308a-0808-11e1-bff3-83d5d70a461f",
      "links" : [
        {
          "rel" : "self",
          "href" : "http://servers.api.openstack.org/v1.1/1234/OS-
BS/volumes/4512308a-0808-11e1-bff3-83d5d70a461f"
        },
        {
          "rel" : "bookmark",
          "href" : "http://servers.api.openstack.org/1234/OS-BS/
volumes/4512308a-0808-11e1-bff3-83d5d70a461f"
        }
      ]
    }
  ]
}

```

### 2.3.1.1.2. Detailed List Volumes

Verb	URI	Description
GET	/ {tenantId} /OS-BS/volumes/ detail?changes-since=dateTime& type=UUID&backup=UUID& name=string&marker=UUID& limit=int&size=unsignedInt& status=ExtensibleVolumeStatus	List all volumes (all details).

Normal Response Code(s): 200, 203

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413)

A detailed volume list.

**Table 2.2. Detailed List Volumes Request Parameters**

Name	Style	Type	Description
changes-since	query	DateTime	The changes-since parameter is optional.
type	query	UUID	The type parameter is optional.
backup	query	UUID	The backup parameter is optional.
name	query	String	The name parameter is optional.
marker	query	UUID	The marker parameter is optional.
limit	query	Int	The limit parameter is optional.
size	query	UnsignedInt	The size parameter is optional.
status	query	ExtensibleVolumeStatus	The status parameter is optional.
tenantId	template	String	A unique identifier for the tenant who owns the resource.

### Example 2.3. Detailed List Volumes Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<volumes xmlns="http://docs.openstack.org/servers/api/ext/volumes/v1.0"
  xmlns:atom="http://www.w3.org/2005/Atom">
  <volume id="15e59938-07d5-11e1-90e3-e3dffe0c5983"
    name="Work"
    typeId="35e1c8d2-e477-11e0-93ed-4f2505dfa32e"
    created="2011-09-09T00:00:00Z"
    attached="2011-11-11T00:00:00Z"
    tenantId="27765"
    userId="joeuser"
    size="1024"
    status="ATTACHED">
    <attachments>
      <server id="52415800-8b69-11e0-9b19-734f6af67565">
        <atom:link
          rel="self"
          href="http://servers.api.openstack.org/v1.1/1234/servers/
52415800-8b69-11e0-9b19-734f6af67565"/>
        <atom:link
          rel="bookmark"
          href="http://servers.api.openstack.org/1234/servers/
52415800-8b69-11e0-9b19-734f6af67565"/>
      </server>
    </attachments>
  </volume>
</volumes>
```



```

</attachments>
<metadata>
  <meta key="Type">Work Directory</meta>
  <meta key="Owner">Joe User</meta>
</metadata>
<atom:link
  rel="self"
  href="http://servers.api.openstack.org/v1.1/1234/OS-BS/volumes/
15e59938-07d5-11e1-90e3-e3dffe0c5983"/>
<atom:link
  rel="bookmark"
  href="http://servers.api.openstack.org/1234/OS-BS/volumes/
15e59938-07d5-11e1-90e3-e3dffe0c5983"/>
</volume>
<volume id="4512308a-0808-11e1-bff3-83d5d70a461f"
  name="Home"
  typeId="35e1c8d2-e477-11e0-93ed-4f2505dfa32e"
  created="2011-09-15T00:00:00Z"
  tenantId="27765"
  userId="joeuser"
  size="1024"
  status="ATTACHING">
<attachments>
  <server id="52415800-8b69-11e0-9b19-734f6af67565">
    <atom:link
      rel="self"
      href="http://servers.api.openstack.org/v1.1/1234/servers/
52415800-8b69-11e0-9b19-734f6af67565"/>
    <atom:link
      rel="bookmark"
      href="http://servers.api.openstack.org/1234/servers/
52415800-8b69-11e0-9b19-734f6af67565"/>
  </server>
</attachments>
<metadata>
  <meta key="Type">Home Directory</meta>
  <meta key="Owner">Joe User</meta>
</metadata>
<atom:link
  rel="self"
  href="http://servers.api.openstack.org/v1.1/1234/OS-BS/volumes/
4512308a-0808-11e1-bff3-83d5d70a461f"/>
<atom:link
  rel="bookmark"
  href="http://servers.api.openstack.org/1234/OS-BS/volumes/
4512308a-0808-11e1-bff3-83d5d70a461f"/>
</volume>
</volumes>

```

### Example 2.4. Detailed List Volumes Response: JSON

```

{
  "OS-BS:volumes" : [
    {
      "name" : "Work",
      "id" : "15e59938-07d5-11e1-90e3-e3dffe0c5983",
      "typeId" : "35e1c8d2-e477-11e0-93ed-4f2505dfa32e",
      "created" : "2011-09-09T00:00:00Z",
      "attached" : "2011-11-11T00:00:00Z",

```

```

    "tenantId" : "27765",
    "userId" : "joeuser",
    "size" : 1024,
    "status" : "ATTACHED",
    "attachments" : [
      {
        "id" : "52415800-8b69-11e0-9b19-734f6af67565",
        "links" : [
          {
            "rel" : "self",
            "href" : "http://servers.api.openstack.org/v1.1/1234/servers/52415800-8b69-11e0-9b19-734f6af67565"
          },
          {
            "rel" : "bookmark",
            "href" : "http://servers.api.openstack.org/1234/servers/52415800-8b69-11e0-9b19-734f6af67565"
          }
        ]
      }
    ],
    "metadata" : {
      "Type" : "Work Directory",
      "Owner" : "Joe User"
    },
    "links" : [
      {
        "rel" : "self",
        "href" : "http://servers.api.openstack.org/v1.1/1234/OS-BS/volumes/15e59938-07d5-11e1-90e3-e3dffe0c5983"
      },
      {
        "rel" : "bookmark",
        "href" : "http://servers.api.openstack.org/1234/OS-BS/volumes/15e59938-07d5-11e1-90e3-e3dffe0c5983"
      }
    ]
  },
  {
    "name" : "Home",
    "id" : "4512308a-0808-11e1-bff3-83d5d70a461f",
    "typeId" : "35e1c8d2-e477-11e0-93ed-4f2505dfa32e",
    "created" : "2011-09-15T00:00:00Z",
    "tenantId" : "27765",
    "userId" : "joeuser",
    "size" : 1024,
    "status" : "ATTACHING",
    "attachments" : [
      {
        "id" : "52415800-8b69-11e0-9b19-734f6af67565",
        "links" : [
          {
            "rel" : "self",
            "href" : "http://servers.api.openstack.org/v1.1/1234/servers/52415800-8b69-11e0-9b19-734f6af67565"
          },
          {
            "rel" : "bookmark",
            "href" : "http://servers.api.openstack.org/1234/servers/52415800-8b69-11e0-9b19-734f6af67565"
          }
        ]
      }
    ]
  }
]

```

```
    }
  ],
  },
  "metadata" : {
    "Type" : "Home Directory",
    "Owner" : "Joe User"
  },
  "links" : [
    {
      "rel" : "self",
      "href" : "http://servers.api.openstack.org/v1.1/1234/OS-
BS/volumes/4512308a-0808-11e1-bff3-83d5d70a461f"
    },
    {
      "rel" : "bookmark",
      "href" : "http://servers.api.openstack.org/1234/OS-BS/
volumes/4512308a-0808-11e1-bff3-83d5d70a461f"
    }
  ]
}
]
```

### 2.3.1.2. Volume Opreations

Verb	URI	Description
POST	/ {tenantId} /OS-BS/volumes	Creates a new volume.
PUT	/ {tenantId} /OS-BS/volumes/{id}	Update the specified volume's name.
GET	/ {tenantId} /OS-BS/volumes/{id}	List details of the specified volume.
DELETE	/ {tenantId} /OS-BS/volumes/{id}	Delete the volume.

### 2.3.1.2.1. Create Volume

Verb	URI	Description
POST	/tenantId}/OS-BS/volumes	Creates a new volume.

Normal Response Code(s): 202

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404), badMediaType (415)

Status Transition:           CREATING → AVAILABLE  
                                   CREATING → ERROR (on error)

The operation creates a new volume. The `size` attribute is optional when creating a volume from a backup by supplying the `backupId`. If a name is not supplied a random name may be generated by the system. The progress of the request can be checked by performing subsequent **GETs** on the URL returned via the `Location` header and available via the `self` link in the representation. The representation also contains a `progress` attribute which indicates percent completion.

Custom volume metadata can also be supplied at create time. The maximum size of the metadata key and value is 255 bytes each. The maximum number of key-value pairs that can be supplied per volume is determined by the compute provider and may be queried via the OS-BS:maxVolumeMeta absolute limit.

**Table 2.3. Create Volume Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.

### Example 2.5. Create Volume Request: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<volume xmlns="http://docs.openstack.org/servers/api/ext/volumes/v1.0"
  name="Work"
  typeId="35e1c8d2-e477-11e0-93ed-4f2505dfa32e"
  size="1024">
  <metadata>
    <meta key="Type">Work</meta>
  </metadata>
</volume>
```

### Example 2.6. Create Volume Request: JSON

```
{
  "OS-BS:volume" : {
    "name" : "Work",
    "typeId" : "35e1c8d2-e477-11e0-93ed-4f2505dfa32e",
    "size" : 1024,
    "metadata" : {
      "Type" : "Work"
    }
  }
}
```

```

    }
  }
}

```

**Table 2.4. Create Volume Response Parameters**

Name	Style	Type	Description
Location	header	AnyURI	<p>The full URL to the newly created volume is returned via the <code>Location</code> header. This URL is also available via the <code>self</code> link in the volume representation.</p> <p>The <code>Location</code> header should always be supplied.</p>

**Example 2.7. Create Volume Response: XML**

```

<?xml version="1.0" encoding="UTF-8"?>

<volume xmlns="http://docs.openstack.org/servers/api/ext/volumes/v1.0"
  xmlns:atom="http://www.w3.org/2005/Atom"
  id="15e59938-07d5-11e1-90e3-e3dffe0c5983"
  name="Work"
  typeId="35e1c8d2-e477-11e0-93ed-4f2505dfa32e"
  created="2011-09-09T00:00:00Z"
  tenantId="27765"
  userId="joeuser"
  size="1024"
  progress="0"
  status="CREATING">
  <metadata>
    <meta key="Type">Work</meta>
  </metadata>
  <atom:link
    rel="self"
    href="http://servers.api.openstack.org/v1.1/1234/OS-BS/volumes/
15e59938-07d5-11e1-90e3-e3dffe0c5983"/>
  <atom:link
    rel="bookmark"
    href="http://servers.api.openstack.org/1234/OS-BS/volumes/
15e59938-07d5-11e1-90e3-e3dffe0c5983"/>
</volume>

```

**Example 2.8. Create Volume Response: JSON**

```

{
  "OS-BS:volume" : {
    "name" : "Work",
    "id" : "15e59938-07d5-11e1-90e3-e3dffe0c5983",
    "typeId" : "35e1c8d2-e477-11e0-93ed-4f2505dfa32e",
    "created" : "2011-09-09T00:00:00Z",
    "tenantId" : "27765",
    "userId" : "joeuser",
    "progress" : 0,
    "size" : 1024,
    "status" : "CREATING",
    "metadata" : {
      "Type" : "Work"
    },
  },
}

```

```
    "links" : [  
      {  
        "rel" : "self",  
        "href" : "http://servers.api.openstack.org/v1.1/1234/OS-BS/  
volumes/15e59938-07d5-11e1-90e3-e3dffe0c5983"  
      },  
      {  
        "rel" : "bookmark",  
        "href" : "http://servers.api.openstack.org/1234/OS-BS/volumes/  
15e59938-07d5-11e1-90e3-e3dffe0c5983"  
      }  
    ]  
  }  
}
```

### 2.3.1.2.2. Update Volume Name

Verb	URI	Description
PUT	/ {tenantId} /OS-BS/volumes/{id}	Update the specified volume's name.

Normal Response Code(s): 200

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404), badMediaType (415)

Status Transition: AVAILABLE → AVAILABLE

The operation updates the volume's name. Only the volume's name is updateable, other fields are ignored.

**Table 2.5. Update Volume Name Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the volume..

#### Example 2.9. Update Volume Name Request: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<volume xmlns="http://docs.openstack.org/servers/api/ext/volumes/v1.0"
  name="New Work"/>
```

#### Example 2.10. Update Volume Name Request: JSON

```
{
  "OS-BS:volume" : {
    "name" : "New Work"
  }
}
```

#### Example 2.11. Update Volume Name Response: XML

```
<volume xmlns="http://docs.openstack.org/servers/api/ext/volumes/v1.0"
  xmlns:atom="http://www.w3.org/2005/Atom"
  id="15e59938-07d5-11e1-90e3-e3dffe0c5983"
  name="New Work"
  typeId="35e1c8d2-e477-11e0-93ed-4f2505dfa32e"
  updated="2011-11-15T00:00:00Z"
  created="2011-09-09T00:00:00Z"
  attached="2011-11-11T00:00:00Z"
  tenantId="27765"
  userId="joeuser"
  size="1024"
  status="ATTACHED">
  <attachments>
```

```

    <server id="52415800-8b69-11e0-9b19-734f6af67565">
      <atom:link
        rel="self"
        href="http://servers.api.openstack.org/v1.1/1234/servers/
52415800-8b69-11e0-9b19-734f6af67565"/>
      <atom:link
        rel="bookmark"
        href="http://servers.api.openstack.org/1234/servers/
52415800-8b69-11e0-9b19-734f6af67565"/>
    </server>
  </attachments>
  <metadata>
    <meta key="Type">Work Directory</meta>
    <meta key="Owner">Joe User</meta>
  </metadata>
  <atom:link
    rel="self"
    href="http://servers.api.openstack.org/v1.1/1234/OS-BS/volumes/
15e59938-07d5-11e1-90e3-e3dffe0c5983"/>
  <atom:link
    rel="bookmark"
    href="http://servers.api.openstack.org/1234/OS-BS/volumes/
15e59938-07d5-11e1-90e3-e3dffe0c5983"/>
</volume>

```

### Example 2.12. Update Volume Name Response: JSON

```

{
  "OS-BS:volume" : {
    "name" : "New Work",
    "id" : "15e59938-07d5-11e1-90e3-e3dffe0c5983",
    "typeId" : "35e1c8d2-e477-11e0-93ed-4f2505dfa32e",
    "updated" : "2011-11-15T00:00:00Z",
    "created" : "2011-09-09T00:00:00Z",
    "attached" : "2011-11-11T00:00:00Z",
    "tenantId" : "27765",
    "userId" : "joeuser",
    "size" : 1024,
    "status" : "ATTACHED",
    "attachments" : [
      {
        "id" : "52415800-8b69-11e0-9b19-734f6af67565",
        "links" : [
          {
            "rel" : "self",
            "href" : "http://servers.api.openstack.org/v1.1/1234/
servers/52415800-8b69-11e0-9b19-734f6af67565"
          },
          {
            "rel" : "bookmark",
            "href" : "http://servers.api.openstack.org/1234/
servers/52415800-8b69-11e0-9b19-734f6af67565"
          }
        ]
      }
    ],
    "metadata" : {
      "Type" : "Work Directory",
      "Owner" : "Joe User"
    }
  }
}

```



```
    },  
    "links" : [  
      {  
        "rel" : "self",  
        "href" : "http://servers.api.openstack.org/v1.1/1234/OS-BS/  
volumes/15e59938-07d5-11e1-90e3-e3dffe0c5983"  
      },  
      {  
        "rel" : "bookmark",  
        "href" : "http://servers.api.openstack.org/1234/OS-BS/volumes/  
15e59938-07d5-11e1-90e3-e3dffe0c5983"  
      }  
    ]  
  }  
}
```

### 2.3.1.2.3. Get Volume Details

Verb	URI	Description
GET	/ {tenantId} /OS-BS/volumes/{id}	List details of the specified volume.

Normal Response Code(s): 200, 203

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404)

**Table 2.6. Get Volume Details Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the volume..

This operation does not require a request body.

### Example 2.13. Get Volume Details Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<volume xmlns="http://docs.openstack.org/servers/api/ext/volumes/v1.0"
  xmlns:atom="http://www.w3.org/2005/Atom"
  id="15e59938-07d5-11e1-90e3-e3dffe0c5983"
  name="Work"
  typeId="35e1c8d2-e477-11e0-93ed-4f2505dfa32e"
  created="2011-09-09T00:00:00Z"
  attached="2011-11-11T00:00:00Z"
  tenantId="27765"
  userId="joeuser"
  size="1024"
  status="ATTACHED">
  <attachments>
    <server id="52415800-8b69-11e0-9b19-734f6af67565">
      <atom:link
        rel="self"
        href="http://servers.api.openstack.org/v1.1/1234/servers/
52415800-8b69-11e0-9b19-734f6af67565"/>
      <atom:link
        rel="bookmark"
        href="http://servers.api.openstack.org/1234/servers/
52415800-8b69-11e0-9b19-734f6af67565"/>
    </server>
  </attachments>
  <metadata>
    <meta key="Type">Work Directory</meta>
    <meta key="Owner">Joe User</meta>
  </metadata>
  <atom:link
    rel="self"
    href="http://servers.api.openstack.org/v1.1/1234/OS-BS/volumes/
15e59938-07d5-11e1-90e3-e3dffe0c5983"/>
  <atom:link
```

```

        rel="bookmark"
        href="http://servers.api.openstack.org/1234/OS-BS/volumes/
15e59938-07d5-11e1-90e3-e3dffe0c5983"/>
</volume>

```

### Example 2.14. Get Volume Details Response: JSON

```

{
  "OS-BS:volume" : {
    "name" : "Work",
    "id" : "15e59938-07d5-11e1-90e3-e3dffe0c5983",
    "typeId" : "35e1c8d2-e477-11e0-93ed-4f2505dfa32e",
    "created" : "2011-09-09T00:00:00Z",
    "attached" : "2011-11-11T00:00:00Z",
    "tenantId" : "27765",
    "userId" : "joeuser",
    "size" : 1024,
    "status" : "ATTACHED",
    "attachments" : [
      {
        "id" : "52415800-8b69-11e0-9b19-734f6af67565",
        "links" : [
          {
            "rel" : "self",
            "href" : "http://servers.api.openstack.org/v1.1/1234/
servers/52415800-8b69-11e0-9b19-734f6af67565"
          },
          {
            "rel" : "bookmark",
            "href" : "http://servers.api.openstack.org/1234/
servers/52415800-8b69-11e0-9b19-734f6af67565"
          }
        ]
      }
    ],
    "metadata" : {
      "Type" : "Work Directory",
      "Owner" : "Joe User"
    },
    "links" : [
      {
        "rel" : "self",
        "href" : "http://servers.api.openstack.org/v1.1/1234/OS-BS/
volumes/15e59938-07d5-11e1-90e3-e3dffe0c5983"
      },
      {
        "rel" : "bookmark",
        "href" : "http://servers.api.openstack.org/1234/OS-BS/volumes/
15e59938-07d5-11e1-90e3-e3dffe0c5983"
      }
    ]
  }
}

```

### 2.3.1.2.4. Delete Volume

Verb	URI	Description
<b>DELETE</b>	<code>/ {tenantId} /OS-BS/volumes/ {id}</code>	Delete the volume.

Normal Response Code(s): 204

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404)

Status Transition: AVAILABLE → DELETING → DELETED  
 AVAILABLE → DELETING → ERROR (on error)

Deletes the volume. The volume must be unattached and AVAILABLE on in ERROR before it can be deleted.

**Table 2.7. Delete Volume Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the volume..

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

### 2.3.1.3. Volume Metadata

The following operations allow access to volume metadata.

Verb	URI	Description
<b>PUT</b>	<code>/ {tenantId} /OS-BS/volumes/ {id} / metadata</code>	Set metadata.
<b>POST</b>	<code>/ {tenantId} /OS-BS/volumes/ {id} / metadata</code>	Update metadata items.
<b>GET</b>	<code>/ {tenantId} /OS-BS/volumes/ {id} / metadata</code>	List metadata associated with the resource.
<b>GET</b>	<code>/ {tenantId} /OS-BS/volumes/ {id} / metadata/ {key}</code>	Get a metadata item associated with a resource.
<b>PUT</b>	<code>/ {tenantId} /OS-BS/volumes/ {id} / metadata/ {key}</code>	Set metadata item.
<b>DELETE</b>	<code>/ {tenantId} /OS-BS/volumes/ {id} / metadata/ {key}</code>	Delete a metadata item.

### 2.3.1.3.1. Create or Replace Metadata

Verb	URI	Description
PUT	<code>//{tenantId}/OS-BS/volumes/{id}/metadata</code>	Set metadata.

Normal Response Code(s): 200

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404), badMediaType (415), buildInProgress (409)

Sets metadata for the resource. The metadata items set on the resource will be exactly those specified on the requests. Existing metadata items are replaced with the ones provided in the request. Metadata items, set previously, but not specified in the request will be removed.

An overLimit (413) fault may be thrown if the maximum number of metadata items is exceeded.

**Table 2.8. Create or Replace Metadata Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the volume..

#### Example 2.15. Create or Replace Metadata Request: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<metadata xmlns="http://docs.openstack.org/compute/api/v1.1">
  <meta key="Label">Web</meta>
  <meta key="Version">2.1</meta>
</metadata>
```

#### Example 2.16. Create or Replace Metadata Request: JSON

```
{
  "metadata" : {
    "Label" : "Web",
    "Version" : "2.1"
  }
}
```

#### Example 2.17. Create or Replace Metadata Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<metadata xmlns="http://docs.openstack.org/compute/api/v1.1">
  <meta key="Label">Web</meta>
  <meta key="Version">2.1</meta>
</metadata>
```

### Example 2.18. Create or Replace Metadata Response: JSON

```
<?xml version="1.0" encoding="UTF-8"?>

<metadata xmlns="http://docs.openstack.org/compute/api/v1.1">
  <meta key="Label">Web</meta>
  <meta key="Version">2.1</meta>
</metadata>
```

### 2.3.1.3.2. Update Metadata

Verb	URI	Description
POST	<code>//{tenantId}/OS-BS/volumes/{id}/metadata</code>	Update metadata items.

Normal Response Code(s): 200

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404), badMediaType (415), buildInProgress (409)

Updates resource metadata. Updates will replace existing metadata items with the same key. Items not explicitly mentioned in the request will not be modified. An overLimit (413) fault may be thrown if the operation causes the maximum number of metadata items to be exceeded.

**Table 2.9. Update Metadata Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the volume..

#### Example 2.19. Update Metadata Request: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<metadata xmlns="http://docs.openstack.org/compute/api/v1.1">
  <meta key="Label">Web2</meta>
</metadata>
```

#### Example 2.20. Update Metadata Request: JSON

```
{
  "metadata" : {
    "Label" : "Web2"
  }
}
```

#### Example 2.21. Update Metadata Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<metadata xmlns="http://docs.openstack.org/compute/api/v1.1">
  <meta key="Label">Web2</meta>
  <meta key="Version">2.1</meta>
</metadata>
```

#### Example 2.22. Update Metadata Response: JSON

```
{
  "metadata" : {
    "Label" : "Web2",
    "Version" : "2.1"
  }
}
```

```
    }  
}
```



### 2.3.1.3.3. List Metadata

Verb	URI	Description
GET	<code>/ {tenantId} /OS-BS/volumes/{id} / metadata</code>	List metadata associated with the resource.

Normal Response Code(s): 200, 203

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404)

List metadata associated with the resource.

**Table 2.10. List Metadata Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the volume..

This operation does not require a request body.

#### Example 2.23. List Metadata Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<metadata xmlns="http://docs.openstack.org/compute/api/v1.1">
  <meta key="Label">Web</meta>
  <meta key="Version">2.1</meta>
</metadata>
```

#### Example 2.24. List Metadata Response: JSON

```
{
  "metadata" : {
    "Label" : "Web",
    "Version" : "2.1"
  }
}
```

### 2.3.1.3.4. Get Metadata Item

Verb	URI	Description
GET	<code>/ {tenantId} /OS-BS/volumes/{id} / metadata/{key}</code>	Get a metadata item associated with a resource.

Normal Response Code(s): 200, 203

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404)

Retrieves a single metadata item by key.

**Table 2.11. Get Metadata Item Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the volume..
key	template	MetadataKey	A string no longer than 255 characters.

This operation does not require a request body.

### Example 2.25. Get Metadata Item Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>
<meta
  xmlns="http://docs.openstack.org/compute/api/v1.1"
  key="Label">Web</meta>
```

### Example 2.26. Get Metadata Item Response: JSON

```
{
  "meta" : {
    "Label" : "Web"
  }
}
```

### 2.3.1.3.5. Create or Update Metadata Item

Verb	URI	Description
PUT	<code>/ {tenantId} /OS-BS/volumes/{id} / metadata/{key}</code>	Set metadata item.

Normal Response Code(s): 200

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404), badMediaType (415), buildInProgress (409)

Sets a metadata item by its key. An overLimit (413) fault may be thrown if the operation causes the maximum number of metadata items to be exceeded.

**Table 2.12. Create or Update Metadata Item Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the volume..
key	template	MetadataKey	A string no longer than 255 characters.

#### Example 2.27. Create or Update Metadata Item Request: XML

```
<?xml version="1.0" encoding="UTF-8"?>
<meta
  xmlns="http://docs.openstack.org/compute/api/v1.1"
  key="Label">Web</meta>
```

#### Example 2.28. Create or Update Metadata Item Request: JSON

```
{
  "meta" : {
    "Label" : "Web"
  }
}
```

#### Example 2.29. Create or Update Metadata Item Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>
<meta
  xmlns="http://docs.openstack.org/compute/api/v1.1"
  key="Label">Web</meta>
```

#### Example 2.30. Create or Update Metadata Item Response: JSON

```
{
  "meta" : {
    "Label" : "Web"
  }
}
```

### 2.3.1.3.6. Delete Metadata Item

Verb	URI	Description
<b>DELETE</b>	<code>/ {tenantId} /OS-BS/volumes/{id} / metadata/{key}</code>	Delete a metadata item.

Normal Response Code(s): 204

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404), buildInProgress (409)

Delete a metadata item.

**Table 2.13. Delete Metadata Item Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the volume..
key	template	MetadataKey	A string no longer than 255 characters.

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

## 2.3.2. Backups

### 2.3.2.1. Backup Lists

Verb	URI	Description
<b>GET</b>	<code>/ {tenantId} /OS-BS/backups?changes-since=dateTime&amp;volume=UUID&amp;name=string&amp;marker=UUID&amp;limit=int&amp;size=unsignedInt&amp;status=ExtensibleBackupStatus</code>	List all backups (IDs, names, links).
<b>GET</b>	<code>/ {tenantId} /OS-BS/backups/detail?changes-since=dateTime&amp;volume=UUID&amp;name=string&amp;marker=UUID&amp;limit=int&amp;size=unsignedInt&amp;status=ExtensibleBackupStatus</code>	List all backups (all details).

### 2.3.2.1.1. List Backups

Verb	URI	Description
GET	<code>//{tenantId}/OS-BS/backups?changes-since=dateTime&amp;volume=UUID&amp;name=string&amp;marker=UUID&amp;limit=int&amp;size=unsignedInt&amp;status=ExtensibleBackupStatus</code>	List all backups (IDs, names, links).

Normal Response Code(s): 200, 203

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413)

A list of backups. Each backup contains IDs, names, and links – other attributes are omitted.

**Table 2.14. List Backups Request Parameters**

Name	Style	Type	Description
changes-since	query	DateTime	The changes-since parameter is optional.
volume	query	UUID	The volume parameter is optional.
name	query	String	The name parameter is optional.
marker	query	UUID	The marker parameter is optional.
limit	query	Int	The limit parameter is optional.
size	query	UnsignedInt	The size parameter is optional.
status	query	ExtensibleBackupStatus	The status parameter is optional.
tenantId	template	String	A unique identifier for the tenant who owns the resource.

### Example 2.31. List Backups Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<backups xmlns="http://docs.openstack.org/servers/api/ext/volumes/v1.0"
  xmlns:atom="http://www.w3.org/2005/Atom">
  <backup id="f9822734-07d4-11e1-a16e-2396601839a1"
    name="My Backup">
    <atom:link
      rel="self"
      href="http://servers.api.openstack.org/v1.1/1234/OS-BS/backups/
f9822734-07d4-11e1-a16e-2396601839a1"/>
    <atom:link
      rel="bookmark"
      href="http://servers.api.openstack.org/1234/OS-BS/backups/
f9822734-07d4-11e1-a16e-2396601839a1"/>
  </backup>
  <backup id="ab7cbf2c-07e8-11e1-98d5-9b76347df0b3"
    name="Original">
    <atom:link
      rel="self"
      href="http://servers.api.openstack.org/v1.1/1234/OS-BS/backups/
ab7cbf2c-07e8-11e1-98d5-9b76347df0b3"/>
    <atom:link
      rel="bookmark"
      href="http://servers.api.openstack.org/1234/OS-BS/backups/
ab7cbf2c-07e8-11e1-98d5-9b76347df0b3"/>
  </backup>
</backups>
```

```
</backup>
</backups>
```

### Example 2.32. List Backups Response: JSON

```
{
  "OS-BS:backups" : [
    {
      "name" : "My Backup",
      "id" : "f9822734-07d4-11e1-a16e-2396601839a1",
      "links" : [
        {
          "rel" : "self",
          "href" : "http://servers.api.openstack.org/v1.1/1234/OS-
BS/backups/f9822734-07d4-11e1-a16e-2396601839a1"
        },
        {
          "rel" : "bookmark",
          "href" : "http://servers.api.openstack.org/1234/OS-BS/
backups/f9822734-07d4-11e1-a16e-2396601839a1"
        }
      ]
    },
    {
      "name" : "Original",
      "id" : "ab7cbf2c-07e8-11e1-98d5-9b76347df0b3",
      "links" : [
        {
          "rel" : "self",
          "href" : "http://servers.api.openstack.org/v1.1/1234/OS-
BS/backups/ab7cbf2c-07e8-11e1-98d5-9b76347df0b3"
        },
        {
          "rel" : "bookmark",
          "href" : "http://servers.api.openstack.org/1234/OS-BS/
backups/ab7cbf2c-07e8-11e1-98d5-9b76347df0b3"
        }
      ]
    }
  ]
}
```

### 2.3.2.1.2. Detailed List Backups

Verb	URI	Description
GET	<code>//{tenantId}/OS-BS/backups/detail?changes-since=dateTime&amp;volume=UUID&amp;name=string&amp;marker=UUID&amp;limit=int&amp;size=unsignedInt&amp;status=ExtensibleBackupStatus</code>	List all backups (all details).

Normal Response Code(s): 200, 203

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413)

A detailed backup list.

**Table 2.15. Detailed List Backups Request Parameters**

Name	Style	Type	Description
changes-since	query	DateTime	The changes-since parameter is optional.
volume	query	UUID	The volume parameter is optional.
name	query	String	The name parameter is optional.
marker	query	UUID	The marker parameter is optional.
limit	query	Int	The limit parameter is optional.
size	query	UnsignedInt	The size parameter is optional.
status	query	ExtensibleBackupStatus	The status parameter is optional.
tenantId	template	String	A unique identifier for the tenant who owns the resource.

### Example 2.33. Detailed List Backups Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<backups xmlns="http://docs.openstack.org/servers/api/ext/volumes/v1.0"
  xmlns:atom="http://www.w3.org/2005/Atom">
  <backup id="f9822734-07d4-11e1-a16e-2396601839a1"
    name="My Backup"
    volumeId="15e59938-07d5-11e1-90e3-e3dffe0c5983"
    created="2011-09-09T00:00:00Z"
    tenantId="27765"
    userId="joeuser"
    size="1024"
    status="AVAILABLE">
    <metadata>
      <meta key="BackupType">Yearly</meta>
      <meta key="ImageVersion">2.1</meta>
    </metadata>
    <atom:link
      rel="self"
      href="http://servers.api.openstack.org/v1.1/1234/OS-BS/backups/
f9822734-07d4-11e1-a16e-2396601839a1"/>
    <atom:link
      rel="bookmark"
      href="http://servers.api.openstack.org/1234/OS-BS/backups/
f9822734-07d4-11e1-a16e-2396601839a1"/>
  </backup>
</backups>
```

```

</backup>
<backup id="ab7cbf2c-07e8-11e1-98d5-9b76347df0b3"
      name="Original"
      volumeId="c04898cc-07e8-11e1-94ef-3f8bfa4fdaa6"
      created="2011-08-17T00:00:00Z"
      tenantId="27765"
      userId="joeuser"
      size="1024"
      progress="55"
      status="SAVING">
  <atom:link
    rel="self"
    href="http://servers.api.openstack.org/v1.1/1234/OS-BS/backups/
ab7cbf2c-07e8-11e1-98d5-9b76347df0b3"/>
  <atom:link
    rel="bookmark"
    href="http://servers.api.openstack.org/1234/OS-BS/backups/
ab7cbf2c-07e8-11e1-98d5-9b76347df0b3"/>
</backup>
</backups>

```

### Example 2.34. Detailed List Backups Response: JSON

```

{
  "OS-BS:backups" : [
    {
      "name" : "My Backup",
      "id" : "f9822734-07d4-11e1-a16e-2396601839a1",
      "volumeId" : "15e59938-07d5-11e1-90e3-e3dffe0c5983",
      "created" : "2011-09-09T00:00:00Z",
      "tenantId" : "27765",
      "userId" : "joeuser",
      "size" : 1024,
      "status" : "AVAILABLE",
      "metadata" : {
        "BackupType" : "Yearly",
        "ImageVersion" : "2.1"
      },
      "links" : [
        {
          "rel" : "self",
          "href" : "http://servers.api.openstack.org/v1.1/1234/OS-
BS/backups/f9822734-07d4-11e1-a16e-2396601839a1"
        },
        {
          "rel" : "bookmark",
          "href" : "http://servers.api.openstack.org/1234/OS-BS/
backups/f9822734-07d4-11e1-a16e-2396601839a1"
        }
      ]
    },
    {
      "name" : "Original",
      "id" : "ab7cbf2c-07e8-11e1-98d5-9b76347df0b3",
      "volumeId" : "c04898cc-07e8-11e1-94ef-3f8bfa4fdaa6",
      "created" : "2011-08-17T00:00:00Z",
      "tenantId" : "27765",
      "userId" : "joeuser",
      "progress" : 55,

```



```
    "size" : 1024,
    "status" : "SAVING",
    "links" : [
      {
        "rel" : "self",
        "href" : "http://servers.api.openstack.org/v1.1/1234/OS-
BS/backups/ab7cbf2c-07e8-11e1-98d5-9b76347df0b3"
      },
      {
        "rel" : "bookmark",
        "href" : "http://servers.api.openstack.org/1234/OS-BS/
backups/ab7cbf2c-07e8-11e1-98d5-9b76347df0b3"
      }
    ]
  }
}
```

### 2.3.2.2. Backup Opreations

Verb	URI	Description
POST	/ {tenantId} /OS-BS/backups	Creates a new backup.
PUT	/ {tenantId} /OS-BS/backups/ {id}	Update the specified backup's name.
GET	/ {tenantId} /OS-BS/backups/ {id}	List details of the specified backup.
DELETE	/ {tenantId} /OS-BS/backups/ {id}	Delete the backup.

### 2.3.2.2.1. Create Backup

Verb	URI	Description
POST	/ {tenantId} /OS-BS/backups	Creates a new backup.

Normal Response Code(s): 202

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404), badMediaType (415)

Status Transition:            SAVING → AVAILABLE  
                                      SAVING → ERROR (on error)

The operation asynchronously creates a backup of an existing volume. The `volumeId` of the volume to backup is required. If a `name` is not supplied a random name may be generated by the system. The progress of the request can be checked by performing subsequent **GETs** on the URL returned via the `Location` header and available via the `self` link in the representation. The representation also contains a `progress` attribute which indicates percent completion.

Custom volume metadata can be supplied at create time. The maximum size of the metadata key and value is 255 bytes each. The maximum number of key-value pairs that can be supplied per volume is determined by the compute provider and may be queried via the OS-BS:maxBackupMeta absolute limit.

**Table 2.16. Create Backup Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.

#### Example 2.35. Create Backup Request: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<backup xmlns="http://docs.openstack.org/servers/api/ext/volumes/v1.0"
  name="My Backup"
  volumeId="15e59938-07d5-11e1-90e3-e3dffe0c5983">
  <metadata>
    <meta key="BackupType">Yearly</meta>
    <meta key="ImageVersion">2.1</meta>
  </metadata>
</backup>
```

#### Example 2.36. Create Backup Request: JSON

```
{
  "OS-BS:backup" : {
    "name" : "My Backup",
    "volumeId" : "15e59938-07d5-11e1-90e3-e3dffe0c5983",
    "metadata" : {
      "BackupType" : "Yearly",
      "ImageVersion" : "2.1"
    }
  }
}
```

```

    }
  }
}

```

**Table 2.17. Create Backup Response Parameters**

Name	Style	Type	Description
Location	header	AnyURI	<p>The full URL to the newly created backup is returned via the <code>Location</code> header. This URL is also available via the <code>self</code> link in the backup representation.</p> <p>The <code>Location</code> header should always be supplied.</p>

**Example 2.37. Create Backup Response: XML**

```

<?xml version="1.0" encoding="UTF-8"?>

<backup xmlns="http://docs.openstack.org/servers/api/ext/volumes/v1.0"
  xmlns:atom="http://www.w3.org/2005/Atom"
  id="f9822734-07d4-11e1-a16e-2396601839a1"
  name="My Backup"
  volumeId="15e59938-07d5-11e1-90e3-e3dffe0c5983"
  created="2011-09-09T00:00:00Z"
  tenantId="27765"
  userId="joeuser"
  size="1024"
  progress="0"
  status="SAVING">
  <metadata>
    <meta key="BackupType">Yearly</meta>
    <meta key="ImageVersion">2.1</meta>
  </metadata>
  <atom:link
    rel="self"
    href="http://servers.api.openstack.org/v1.1/1234/OS-BS/backups/
f9822734-07d4-11e1-a16e-2396601839a1"/>
  <atom:link
    rel="bookmark"
    href="http://servers.api.openstack.org/1234/OS-BS/backups/
f9822734-07d4-11e1-a16e-2396601839a1"/>
</backup>

```

**Example 2.38. Create Backup Response: JSON**

```

{
  "OS-BS:backup" : {
    "name" : "My Backup",
    "id" : "f9822734-07d4-11e1-a16e-2396601839a1",
    "volumeId" : "15e59938-07d5-11e1-90e3-e3dffe0c5983",
    "created" : "2011-09-09T00:00:00Z",
    "tenantId" : "27765",
    "userId" : "joeuser",
    "progress" : 0,
    "size" : 1024,
    "status" : "SAVING",
    "metadata" : {
      "BackupType" : "Yearly",

```

```
        "ImageVersion" : "2.1"
      },
      "links" : [
        {
          "rel" : "self",
          "href" : "http://servers.api.openstack.org/v1.1/1234/OS-BS/
backups/f9822734-07d4-11e1-a16e-2396601839a1"
        },
        {
          "rel" : "bookmark",
          "href" : "http://servers.api.openstack.org/1234/OS-BS/backups/
f9822734-07d4-11e1-a16e-2396601839a1"
        }
      ]
    }
  }
}
```

### 2.3.2.2.2. Update Backup Name

Verb	URI	Description
PUT	<code>/ {tenantId} /OS-BS/backups/{id}</code>	Update the specified backup's name.

Normal Response Code(s): 200

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404), badMediaType (415)

Status Transition: AVAILABLE → AVAILABLE

The operation updates the backup's name. Only the backup's name is updateable, other fields are ignored.

**Table 2.18. Update Backup Name Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the backup.

#### Example 2.39. Update Backup Name Request: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<backup xmlns="http://docs.openstack.org/servers/api/ext/volumes/v1.0"
  name="New backup name"/>
```

#### Example 2.40. Update Backup Name Request: JSON

```
{
  "OS-BS:backup" : {
    "name" : "New backup name"
  }
}
```

#### Example 2.41. Update Backup Name Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<backup xmlns="http://docs.openstack.org/servers/api/ext/volumes/v1.0"
  xmlns:atom="http://www.w3.org/2005/Atom"
  id="f9822734-07d4-11e1-a16e-2396601839a1"
  name="New backup name"
  volumeId="15e59938-07d5-11e1-90e3-e3dffe0c5983"
  created="2011-09-09T00:00:00Z"
  updated="2011-12-12T00:00:00Z"
  tenantId="27765"
  userId="joeuser"
  size="1024"
  status="AVAILABLE">
  <metadata>
```

```

    <meta key="BackupType">Yearly</meta>
    <meta key="ImageVersion">2.1</meta>
  </metadata>
  <atom:link
    rel="self"
    href="http://servers.api.openstack.org/v1.1/1234/OS-BS/backups/
f9822734-07d4-11e1-a16e-2396601839a1"/>
  <atom:link
    rel="bookmark"
    href="http://servers.api.openstack.org/1234/OS-BS/backups/
f9822734-07d4-11e1-a16e-2396601839a1"/>
</backup>

```

### Example 2.42. Update Backup Name Response: JSON

```

{
  "OS-BS:backup" : {
    "name" : "New backup name",
    "id" : "f9822734-07d4-11e1-a16e-2396601839a1",
    "volumeId" : "15e59938-07d5-11e1-90e3-e3dffe0c5983",
    "updated" : "2011-12-12T00:00:00Z",
    "created" : "2011-09-09T00:00:00Z",
    "tenantId" : "27765",
    "userId" : "joeuser",
    "size" : 1024,
    "status" : "AVAILABLE",
    "metadata" : {
      "BackupType" : "Yearly",
      "ImageVersion" : "2.1"
    },
    "links" : [
      {
        "rel" : "self",
        "href" : "http://servers.api.openstack.org/v1.1/1234/OS-BS/
backups/f9822734-07d4-11e1-a16e-2396601839a1"
      },
      {
        "rel" : "bookmark",
        "href" : "http://servers.api.openstack.org/1234/OS-BS/backups/
f9822734-07d4-11e1-a16e-2396601839a1"
      }
    ]
  }
}

```

### 2.3.2.2.3. Get Backup Details

Verb	URI	Description
GET	<code>//{tenantId}/OS-BS/backups/{id}</code>	List details of the specified backup.

Normal Response Code(s): 200, 203

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404)

**Table 2.19. Get Backup Details Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the backup.

This operation does not require a request body.

### Example 2.43. Get Backup Details Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<backup xmlns="http://docs.openstack.org/servers/api/ext/volumes/v1.0"
  xmlns:atom="http://www.w3.org/2005/Atom"
  id="f9822734-07d4-11e1-a16e-2396601839a1"
  name="My Backup"
  volumeId="15e59938-07d5-11e1-90e3-e3dffe0c5983"
  created="2011-09-09T00:00:00Z"
  tenantId="27765"
  userId="joeuser"
  size="1024"
  status="AVAILABLE">
  <metadata>
    <meta key="BackupType">Yearly</meta>
    <meta key="ImageVersion">2.1</meta>
  </metadata>
  <atom:link
    rel="self"
    href="http://servers.api.openstack.org/v1.1/1234/OS-BS/backups/
f9822734-07d4-11e1-a16e-2396601839a1"/>
  <atom:link
    rel="bookmark"
    href="http://servers.api.openstack.org/1234/OS-BS/backups/
f9822734-07d4-11e1-a16e-2396601839a1"/>
</backup>
```

### Example 2.44. Get Backup Details Response: JSON

```
{
  "OS-BS:backup" : {
    "name" : "My Backup",
    "id" : "f9822734-07d4-11e1-a16e-2396601839a1",
    "volumeId" : "15e59938-07d5-11e1-90e3-e3dffe0c5983",
```

```
    "created" : "2011-09-09T00:00:00Z",
    "tenantId" : "27765",
    "userId" : "joeuser",
    "size" : 1024,
    "status" : "AVAILABLE",
    "metadata" : {
      "BackupType" : "Yearly",
      "ImageVersion" : "2.1"
    },
    "links" : [
      {
        "rel" : "self",
        "href" : "http://servers.api.openstack.org/v1.1/1234/OS-BS/
backups/f9822734-07d4-11e1-a16e-2396601839a1"
      },
      {
        "rel" : "bookmark",
        "href" : "http://servers.api.openstack.org/1234/OS-BS/backups/
f9822734-07d4-11e1-a16e-2396601839a1"
      }
    ]
  }
}
```



### 2.3.2.2.4. Delete Backup

Verb	URI	Description
<b>DELETE</b>	<code>/ {tenantId} /OS-BS/backups/ {id}</code>	Delete the backup.

Normal Response Code(s): 204

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404)

Status Transition: AVAILABLE → DELETING → DELETED  
 AVAILABLE → DELETING → ERROR (on error)

Deletes the backup. The backup must be in the AVAILABLE or in ERROR before it can be deleted.

**Table 2.20. Delete Backup Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the backup.

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

### 2.3.2.3. Backup Metadata

The following operations allow access to volume metadata.

Verb	URI	Description
<b>PUT</b>	<code>/ {tenantId} /OS-BS/backups/ {id} / metadata</code>	Set metadata.
<b>POST</b>	<code>/ {tenantId} /OS-BS/backups/ {id} / metadata</code>	Update metadata items.
<b>GET</b>	<code>/ {tenantId} /OS-BS/backups/ {id} / metadata</code>	List metadata associated with the resource.
<b>GET</b>	<code>/ {tenantId} /OS-BS/backups/ {id} / metadata/ {key}</code>	Get a metadata item associated with a resource.
<b>PUT</b>	<code>/ {tenantId} /OS-BS/backups/ {id} / metadata/ {key}</code>	Set metadata item.
<b>DELETE</b>	<code>/ {tenantId} /OS-BS/backups/ {id} / metadata/ {key}</code>	Delete a metadata item.

### 2.3.2.3.1. Create or Replace Metadata

Verb	URI	Description
PUT	<code>/ {tenantId} /OS-BS/backups/ {id} / metadata</code>	Set metadata.

Normal Response Code(s): 200

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404), badMediaType (415), buildInProgress (409)

Sets metadata for the resource. The metadata items set on the resource will be exactly those specified on the requests. Existing metadata items are replaced with the ones provided in the request. Metadata items, set previously, but not specified in the request will be removed.

An overLimit (413) fault may be thrown if the maximum number of metadata items is exceeded.

**Table 2.21. Create or Replace Metadata Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the backup.

#### Example 2.45. Create or Replace Metadata Request: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<metadata xmlns="http://docs.openstack.org/compute/api/v1.1">
  <meta key="Label">Web</meta>
  <meta key="Version">2.1</meta>
</metadata>
```

#### Example 2.46. Create or Replace Metadata Request: JSON

```
{
  "metadata" : {
    "Label" : "Web",
    "Version" : "2.1"
  }
}
```

#### Example 2.47. Create or Replace Metadata Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<metadata xmlns="http://docs.openstack.org/compute/api/v1.1">
  <meta key="Label">Web</meta>
  <meta key="Version">2.1</meta>
</metadata>
```

**Example 2.48. Create or Replace Metadata Response: JSON**

```
<?xml version="1.0" encoding="UTF-8"?>

<metadata xmlns="http://docs.openstack.org/compute/api/v1.1">
  <meta key="Label">Web</meta>
  <meta key="Version">2.1</meta>
</metadata>
```

### 2.3.2.3.2. Update Metadata

Verb	URI	Description
POST	<code>/ {tenantId} /OS-BS/backups/{id} /metadata</code>	Update metadata items.

Normal Response Code(s): 200

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404), badMediaType (415), buildInProgress (409)

Updates resource metadata. Updates will replace existing metadata items with the same key. Items not explicitly mentioned in the request will not be modified. An overLimit (413) fault may be thrown if the operation causes the maximum number of metadata items to be exceeded.

**Table 2.22. Update Metadata Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the backup.

#### Example 2.49. Update Metadata Request: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<metadata xmlns="http://docs.openstack.org/compute/api/v1.1">
  <meta key="Label">Web2</meta>
</metadata>
```

#### Example 2.50. Update Metadata Request: JSON

```
{
  "metadata" : {
    "Label" : "Web2"
  }
}
```

#### Example 2.51. Update Metadata Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<metadata xmlns="http://docs.openstack.org/compute/api/v1.1">
  <meta key="Label">Web2</meta>
  <meta key="Version">2.1</meta>
</metadata>
```

#### Example 2.52. Update Metadata Response: JSON

```
{
  "metadata" : {
    "Label" : "Web2",
    "Version" : "2.1"
  }
}
```

```
    }  
}
```

### 2.3.2.3.3. List Metadata

Verb	URI	Description
GET	<code>/ {tenantId} /OS-BS/backups/{id}/metadata</code>	List metadata associated with the resource.

Normal Response Code(s): 200, 203

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404)

List metadata associated with the resource.

**Table 2.23. List Metadata Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the backup.

This operation does not require a request body.

#### Example 2.53. List Metadata Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<metadata xmlns="http://docs.openstack.org/compute/api/v1.1">
  <meta key="Label">Web</meta>
  <meta key="Version">2.1</meta>
</metadata>
```

#### Example 2.54. List Metadata Response: JSON

```
{
  "metadata" : {
    "Label" : "Web",
    "Version" : "2.1"
  }
}
```

### 2.3.2.3.4. Get Metadata Item

Verb	URI	Description
GET	<code>/ {tenantId} /OS-BS/backups/{id} / metadata/{key}</code>	Get a metadata item associated with a resource.

Normal Response Code(s): 200, 203

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404)

Retrieves a single metadata item by key.

**Table 2.24. Get Metadata Item Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the backup.
key	template	MetadataKey	A string no longer than 255 characters.

This operation does not require a request body.

### Example 2.55. Get Metadata Item Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>
<meta
  xmlns="http://docs.openstack.org/compute/api/v1.1"
  key="Label">Web</meta>
```

### Example 2.56. Get Metadata Item Response: JSON

```
{
  "meta" : {
    "Label" : "Web"
  }
}
```

### 2.3.2.3.5. Create or Update Metadata Item

Verb	URI	Description
PUT	/ {tenantId} /OS-BS/backups/{id} / metadata/{key}	Set metadata item.

Normal Response Code(s): 200

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404), badMediaType (415), buildInProgress (409)

Sets a metadata item by its key. An overLimit (413) fault may be thrown if the operation causes the maximum number of metadata items to be exceeded.

**Table 2.25. Create or Update Metadata Item Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the backup.
key	template	MetadataKey	A string no longer than 255 characters.

#### Example 2.57. Create or Update Metadata Item Request: XML

```
<?xml version="1.0" encoding="UTF-8"?>
<meta
  xmlns="http://docs.openstack.org/compute/api/v1.1"
  key="Label">Web</meta>
```

#### Example 2.58. Create or Update Metadata Item Request: JSON

```
{
  "meta" : {
    "Label" : "Web"
  }
}
```

#### Example 2.59. Create or Update Metadata Item Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>
<meta
  xmlns="http://docs.openstack.org/compute/api/v1.1"
  key="Label">Web</meta>
```

#### Example 2.60. Create or Update Metadata Item Response: JSON

```
{
  "meta" : {
    "Label" : "Web"
  }
}
```



### 2.3.2.3.6. Delete Metadata Item

Verb	URI	Description
<b>DELETE</b>	<code>/ {tenantId} /OS-BS/backups/{id} / metadata/{key}</code>	Delete a metadata item.

Normal Response Code(s): 204

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404), buildInProgress (409)

Delete a metadata item.

**Table 2.26. Delete Metadata Item Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the backup.
key	template	MetadataKey	A string no longer than 255 characters.

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

### 2.3.3. Types

Verb	URI	Description
<b>GET</b>	<code>/ {tenantId} /OS-BS/types? name=string&amp;marker=UUID&amp;limit=int</code>	List all types.
<b>GET</b>	<code>/ {tenantId} /OS-BS/types/{id}</code>	List details of the specified type.

### 2.3.3.1. List Types

Verb	URI	Description
GET	<code>/ {tenantId} /OS-BS/types? name=string&amp;marker=UUID&amp;limit=int</code>	List all types.

Normal Response Code(s): 200, 203

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413)

**Table 2.27. List Types Request Parameters**

Name	Style	Type	Description
name	query	String	The name parameter is optional.
marker	query	UUID	The marker parameter is optional.
limit	query	Int	The limit parameter is optional.
tenantId	template	String	A unique identifier for the tenant who owns the resource.

#### Example 2.61. List Types Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<types xmlns="http://docs.openstack.org/servers/api/ext/volumes/v1.0"
  xmlns:atom="http://www.w3.org/2005/Atom">
  <type id="02d1ab86-e479-11e0-98ab-c7be696d5830" name="NFS">
    <atom:link
      rel="self"
      href="http://servers.api.openstack.org/v1.1/1234/OS-BS/types/
02d1ab86-e479-11e0-98ab-c7be696d5830"/>
    <atom:link
      rel="bookmark"
      href="http://servers.api.openstack.org/1234/OS-BS/types/02d1ab86-
e479-11e0-98ab-c7be696d5830"/>
  </type>
  <type id="35e1c8d2-e477-11e0-93ed-4f2505dfa32e" name="SSD">
    <atom:link
      rel="self"
      href="http://servers.api.openstack.org/v1.1/1234/OS-BS/types/
35e1c8d2-e477-11e0-93ed-4f2505dfa32e"/>
    <atom:link
      rel="bookmark"
      href="http://servers.api.openstack.org/1234/OS-BS/types/35e1c8d2-
e477-11e0-93ed-4f2505dfa32e"/>
  </type>
</types>
```

#### Example 2.62. List Types Response: JSON

```
{
  "OS-BS:types" : [
    {
      "name" : "NFS",
      "id" : "02d1ab86-e479-11e0-98ab-c7be696d5830",
```

```
    "links" : [
      {
        "rel" : "self",
        "href" : "http://servers.api.openstack.org/v1.1/1234/OS-
BS/types/02d1ab86-e479-11e0-98ab-c7be696d5830"
      },
      {
        "rel" : "bookmark",
        "href" : "http://servers.api.openstack.org/1234/OS-BS/
types/02d1ab86-e479-11e0-98ab-c7be696d5830"
      }
    ]
  },
  {
    "name" : "SSD",
    "id" : "35e1c8d2-e477-11e0-93ed-4f2505dfa32e",
    "links" : [
      {
        "rel" : "self",
        "href" : "http://servers.api.openstack.org/v1.1/1234/OS-
BS/types/35e1c8d2-e477-11e0-93ed-4f2505dfa32e"
      },
      {
        "rel" : "bookmark",
        "href" : "http://servers.api.openstack.org/1234/OS-BS/
types/35e1c8d2-e477-11e0-93ed-4f2505dfa32e"
      }
    ]
  }
]
```

### 2.3.3.2. Get Type Details

Verb	URI	Description
GET	<code>/{{tenantId}}/OS-BS/types/{{id}}</code>	List details of the specified type.

Normal Response Code(s): 200, 203

Error Response Code(s): computeFault (400, 500, ...), serviceUnavailable (503), unauthorized (401), forbidden (403), badRequest (400), badMethod (405), overLimit (413), itemNotFound (404)

**Table 2.28. Get Type Details Request Parameters**

Name	Style	Type	Description
tenantId	template	String	A unique identifier for the tenant who owns the resource.
id	template	Uuid	A unique identifier for the type.

This operation does not require a request body.

#### Example 2.63. Get Type Details Response: XML

```
<?xml version="1.0" encoding="UTF-8"?>

<type xmlns="http://docs.openstack.org/servers/api/ext/volumes/v1.0"
      xmlns:atom="http://www.w3.org/2005/Atom"
      id="35e1c8d2-e477-11e0-93ed-4f2505dfa32e" name="SSD">
  <atom:link
    rel="self"
    href="http://servers.api.openstack.org/v1.1/1234/OS-BS/types/35e1c8d2-
e477-11e0-93ed-4f2505dfa32e"/>
  <atom:link
    rel="bookmark"
    href="http://servers.api.openstack.org/1234/OS-BS/types/35e1c8d2-
e477-11e0-93ed-4f2505dfa32e"/>
</type>
```

#### Example 2.64. Get Type Details Response: JSON

```
{
  "OS-BS:type" : {
    "name" : "SSD",
    "id" : "35e1c8d2-e477-11e0-93ed-4f2505dfa32e",
    "links" : [
      {
        "rel" : "self",
        "href" : "http://servers.api.openstack.org/v1.1/1234/OS-BS/
types/35e1c8d2-e477-11e0-93ed-4f2505dfa32e"
      },
      {
        "rel" : "bookmark",
        "href" : "http://servers.api.openstack.org/1234/OS-BS/types/
35e1c8d2-e477-11e0-93ed-4f2505dfa32e"
      }
    ]
  }
}
```

```
}  
  }  
}
```

## 2.4. New Actions

None.

## 2.5. Changes to the OpenStack Compute API

None. This extension simply introduces the ability to create and backup volumes. Please consult the Block Storage Attach/Detach Extensions (OS-BSAD) to allow mounting these volumes to a particular server.