Endpoints and Endpoint Templates

What are Endpoints?

Simply, endpoints are URLs that point to OpenStack services. When you authenticate to Keystone you get back a token which has a service catalog in it. The service catalog is basically a list of the OpenStack services that you have access to and the URLs you can use to get to them; their endpoints.

Here is an example response from Keystone when you authenticate:

```
{
        "access":{
                "token":{
                        "id":"ab48a9efdfedb23ty3494",
                        "expires":"2010-11-01T03:32:15-05:00",
                        "tenant":{
                                "id": "t1000",
                                "name": "My Project"
                        }
                },
                "user":{
                        "id":"u123",
                        "name":"jqsmith",
                        "roles":[{
                                        "id":"100",
                                        "name":"compute:admin"
                                },
                                {
                                        "id":"101",
                                        "name":"object-store:admin",
                                        "tenantId":"t1000"
                                }
                        ],
                        "roles_links":[]
                },
                "serviceCatalog":[{
                                "name":"Nova",
                                "type":"compute",
                                "endpoints":[{
                                                "tenantId":"t1000",
                                                "publicURL":"https://compute.north.host.com/v1/t1
                                                "internalURL":"https://compute.north.internal/v1/
                                                "region":"North",
                                                "versionId":"1",
                                                "versionInfo":"https://compute.north.host.com/v1/
                                                "versionList":"https://compute.north.host.com/"
                                        },
                                        {
                                                "tenantId":"t1000",
                                                "publicURL":"https://compute.north.host.com/v1.1/
                                                "internalURL":"https://compute.north.internal/v1.
                                                "region":"North",
                                                "versionId":"1.1",
                                                "versionInfo":"https://compute.north.host.com/v1.
                                                "versionList":"https://compute.north.host.com/"
                                        }
                                ],
                                "endpoints_links":[]
                        },
                        {
                                "name":"Swift",
                                "name":"Swift",
                                "type":"object-store",
                                "endpoints":[{
                                                "tenantId":"t1000",
                                                "publicURL":"https://storage.north.host.com/v1/t1
                                                "internalURL":"https://storage.north.internal/v1/
                                                "region":"North",
                                                "versionId":"1",
                                                "versionInfo":"https://storage.north.host.com/v1/
                                                "versionList":"https://storage.north.host.com/"
                                        },
                                        {
                                                "tenantId":"t1000",
                                                "publicURL":"https://storage.south.host.com/v1/t1
                                                "internalURL":"https://storage.south.internal/v1/
                                                "region":"South",
                                                "versionId":"1",
                                                "versionInfo":"https://storage.south.host.com/v1/
                                                "versionList":"https://storage.south.host.com/"
                                        }
                                ]
                        },
                        {
                                "name":"DNS-as-a-Service",
                                "type":"dnsextension:dns",
                                "endpoints":[{
                                                "tenantId":"t1000",
                                                "publicURL":"https://dns.host.com/v2.0/t1000",
                                                "versionId":"2.0",
                                                "versionInfo":"https://dns.host.com/v2.0/",
                                                "versionList":"https://dns.host.com/"
                                        }
                                ]
                        }
                ]
        }
}
```

Note the following about this response:

1. There are two endpoints given to the Nova compute service. The only difference between them is the version (1.0 vs. 1.1). This allows for code written to look for the version 1.0 endpoint to still work even after the 1.1 version is released.

2. There are two endpoints for the Swift object-store service. The difference between them is they are in different regions (North and South).

3. Note the DNS service is global; it does not have a Region. Also, since DNS is not a core OpenStack service, the endpoint type is “dnsextension:dns” showing it is coming from an extension to the Keystone service.

4. The Region, Tenant, and versionId are listed under the endpoint. You do not (and should not) have to parse those out of the URL. In fact, they may not be embedded in the URL if the service developer so chooses.

What do the fields in an Endpoint mean?

The schema definition for an endpoint is in endpoints.xsd under keystone/content/common/xsd in the Keystone code repo. The fields are:

id

A unique ID for the endpoint.

type

The OpenStack-registered type (ex. ‘compute’, ‘object-store’, ‘image service’) This can also be extended using the OpenStack Extension mechanism to support non-core services. Extended services will be in the form extension:type. Ex.

> dnsextension:dns

name

This can be anything that the operator of OpenStack chooses. It could be a brand or marketing name (ex. Rackspace Cloud Servers).

region

This is a string that identifies the region where this endpoint exists. Examples are ‘North America’, ‘Europe’, ‘Asia’. Or ‘North’ and ‘South’. Or ‘Data Center 1’, ‘Data Center 2’. The list of regions and what a region means is decided by the operator. The spec treats them as opaque strings.

publicURL

This is the URL to use to access that endpoint over the internet.

internalURL

This is the URL to use to communicate between services. This is genenrally a way to communicate between services over a high bandwidth, low latency, unmetered (free, no bandwidth charges) network. An example would be if you want to access a swift cluster from inside your Nova VMs and want to make sure the communication stays local and does not go over a public network and rack up your bandwidth charges.

adminURL

This is the URL to use to administer the service. In Keystone, this URL is only shown to users with the appropriate rights.

tenantId

If an endpoint is specific to a tenant, the tenantId field identifies the tenant that URL applies to. Some operators include the tenant in the URLs for a service, while others may provide one endpoint and use some other mechanism to identify the tenant. This field is therefore optional. Having this field also means you do not have to parse the URL to identify a tenant if the operator includes it in the URL.

versionId

This identifies the version of the API contract that endpoint supports. While many APIs include the version in the URL (ex: https://compute.host/v1), this field allows you to identify the version without parsing the URL. It therefore also allows operators and service developers to publish endpoints that do not have versions embedded in the URL.

versionInfo

This is the URL to call to get some information on the version. This returns information in this format:

```
{
"version": {
  "id": "v2.0",
  "status": "CURRENT",
  "updated": "2011-01-21T11:33:21-06:00",
  "links": [
    {
      "rel": "self",
      "href": "http://identity.api.openstack.org/v2.0/"
    }, {
      "rel": "describedby",
      "type": "application/pdf",
      "href": "http://docs.openstack.org/identity/api/v2.0/identity-latest.pdf"
    }, {
      "rel": "describedby",
      "type": "application/vnd.sun.wadl+xml",
      "href": "http://docs.openstack.org/identity/api/v2.0/identity.wadl"
    }
  ],
    "media-types": [
      {
        "base": "application/xml",
        "type": "application/vnd.openstack.identity+xml;version=2.0"
      }, {
        "base": "application/json",
        "type": "application/vnd.openstack.identity+json;version=2.0"
      }
    ]
  }
}
```

versionList

This is the URL to call to find out which versions are supported at that endpoint. The response is in this format:

```
{
    "versions":[{
            "id":"v1.0",
            "status":"DEPRECATED",
            "updated":"2009-10-09T11:30:00Z",
            "links":[{
                    "rel":"self",
                    "href":"http://identity.api.openstack.org/v1.0/"
                }
            ]
        },
        {
            "id":"v1.1",
            "status":"CURRENT",
            "updated":"2010-12-12T18:30:02.25Z",
            "links":[{
                    "rel":"self",
                    "href":"http://identity.api.openstack.org/v1.1/"
                }
            ]
        },
        {
            "id":"v2.0",
            "status":"BETA",
            "updated":"2011-05-27T20:22:02.25Z",
            "links":[{
                    "rel":"self",
                    "href":"http://identity.api.openstack.org/v2.0/"
                }
            ]
        }
    ],
    "versions_links":[]
}
```

Here, the response shows that the endpoint supports version 1.0, 1.1, and 2.0. It also shows that 1.0 is in DEPRECTAED status and 2.0 is in BETA.

What are Endpoint Templates?

Endpoint Templates are a way for an administrator to manage endpoints en masse. They provide a way to define Endpoints that apply to many or all tenants without having to a create each endpoint on each tenant manually. Without Endpoint Templates, if I wanted to create Endpoints for each tenant in my OpenStack deployment, I'd have to manually create a bunch of endpoints on each tenant (probably when I created the tenant). And then I'd have to go change them all whenever a service changed versions or I added a new service.

To provide a simpler mechanism to manage endpoints on tenants, Keystone uses Endpoint Templates. I can, for example, define a template with parametrized URLs and set it's *global* to true and that will show up as an endpoint on all the tenants I have. Here is an example:

Define a global Endpoint Template:

```
$ ./keystone-manage endpointTemplates add North nova https://compute.north.example.com/v1/%tenant

The arguments are: object_type action 'region' 'service_name' 'publicURL' 'adminURL' 'internalURL
```

This creates a global endpoint (global means it gets applied to all tenants automatically).

Now, when a user authenticates, they get that endpoint in their service catalog. Here’s an example authentication request for use against tenant 1:

```
$ curl -H "Content-type: application/json" -d '{"auth":{"passwordCredentials":{"username":"joeuse
```

The response is:

```
{
    "access": {
        "serviceCatalog": [
            {
                "endpoints": [
                    {
                        "internalURL": "https://compute.north.example.local",
                        "publicURL": "https://compute.north.example.com/v1/1/",
                        "region": "North"
                    }
                ],
                "name": "nova",
                "type": "compute"
            }
        ],
        "token": {
            "expires": "2012-02-05T00:00:00",
            "id": "887665443383838",
            "tenant": {
                "id": "1",
                "name": "customer-x"
            }
        },
        "user": {
            "id": "1",
            "name": "joeuser",
            "roles": [
                {
                    "id": "3",
                    "name": "Member",
                    "tenantId": "1"
                }
            ]
        }
    }
}
```

Notice the adminURL is not showing (this user is a regular user and does not have rights to see the adminURL) and the tenant ID has been substituted in the URL:

```
"publicURL": "https://compute.north.example.com/v1/1/",
```

This endpoint will show up for all tenants. The OpenStack administrator does not need to create the endpoint manually.

Note

Endpoint Templates are not part of the core Keystone API (but Endpoints are).

What parameters can I use in a Template URL

Currently the only parameterization available is %tenant_id% which gets substituted by the Tenant ID.

Endpoint Template Types: Global or not

When the global flag is set to true on an Endpoint Template, it means it should be available to all tenants. Whenever someone authenticates to a tenant, they will see the Endpoint generated by that template.

When the global flag is not set, the template only shows up when it is added to a tenant manually. To add an endpoint to a tenant manually, you must create the Endpoint and supply the Endpoint Template ID:

Create the Endpoint Template:

```
$ ./keystone-manage endpointTemplates add West nova https://compute.west.example.com/v1/%tenant_i

Note the 0 at the end - this Endpoint Template is not global. So it will not show up for users au
```

Find the Endpoint Template ID:

```
$ ./keystone-manage endpointTemplates list

All EndpointTemplates
id  service type    region  enabled is_global       Public URL      Admin URL
-------------------------------------------------------------------------------
15  nova    compute North   True    True    https://compute.north.example.com/v1/%tenant_id%/
16  nova    compute West    True    False   https://compute.west.example.com/v1/%tenant_id%/
```

Add the Endpoint to the tenant:

```
$ ./keystone-manage endpoint add customer-x 16
```

Now, when the user authenticates, they get the endpoint:

```
{
    "internalURL": "https://compute.west.example.local",
    "publicURL": "https://compute.west.example.com/v1/1/",
    "region": "West"
}
```

Who can see the AdminURL?

Users who have the Keystone `Admin` or `Service Admin` roles will see the AdminURL when they authenticate or when they retrieve token information:

Using an administrator token to authenticate, GET a client token’s endpoints:

```
$ curl -H "X-Auth-Token: 999888777666" http://localhost:35357/v2.0/tokens/887665443383838/endpoin

{
    "endpoints": [
        {
            "adminURL": "https://compute.west.example.corp",
            "id": 6,
            "internalURL": "https://compute.west.example.local",
            "name": "nova",
            "publicURL": "https://compute.west.example.com/v1/1/",
            "region": "West",
            "tenantId": 1,
            "type": "compute"
        }
    ],
    "endpoints_links": [
        {
            "href": "http://127.0.0.1:35357/tokens/887665443383838/endpoints?marker=6&limit=10",
            "rel": "next"
        }
    ]
}
```