Identity Management Using Keystone

Andrew Mallett

LINUX AUTHOR AND TRAINER

@theurbanpenguin www.theurbanpenguin.com

Objectives

Check API version for Keystone

Create service and endpoint records

Manage domains, users, groups, roles

Understand the Keystone configuration

Issue tokens

By Default a PackStack Install is Using V2 API

```
# source keystonerc_admin

# openstack domain list

# openstack endpoint list
```

Check Functionality of the V2 API

Standard V2 Credentials

```
unset OS_SERVICE_TOKEN
export OS_USERNAME=admin
export OS_PASSWORD=Password1
export OS_AUTH_URL=http://192.168.56.5:5000/v2.0
export PS1='[\u@\h \W(keystone_admin)]\$ '
export OS_TENANT_NAME=admin
export OS_REGION_NAME=RegionOne
```

V3 Credentials

```
unset OS_SERVICE_TOKEN
export OS_USERNAME=admin
export OS_PASSWORD=Password1
export OS_AUTH_URL=http://192.168.56.5:5000/
export OS_IDENTITY_API_VERSION=3
export OS_PROJECT_DOMAIN_NAME=Default
export OS_USER_DOMAIN_NAME=Default
export PS1='[\u@\h \W(keystone_admin)]\$ ‘
export OS_TENANT_NAME=admin
export OS_REGION_NAME=RegionOne
```

```
# source keystonerc_adminv3

# openstack domain list

# openstack endpoint list
```

Check Functionality of the V3 API

Verify the Keystone API Operation

Services and Endpoints are used as a Discovery Mechanism in Keystone

```
# openstack service create | delete | show | list

# openstack service create --name “heat” orchestration

# openstack service list

# openstack service show heat

# openstack service delete heat
```

Manage Service Entries

Service entries in Keystone match the names that we give services to their service type. In this example, we create a service named heat of the type orchestration.

public - facing users, the main cloud users

internal - communication between cloud nodes

admin - Reserved for cloud administration. Not Internet facing

Manage Endpoints

In an ideal cloud, our main controller node would have 3 NICs connected to separate networks. Endpoints in the catalog are linked to one or more of these networks

```
# openstack endpoint create \
  --region RegionOne identity public \
  http://192.168.56.5:5000/v3

# openstack endpoint create \
  --region RegionOne identity admin \
  http://192.168.56.5:35357/v3
```

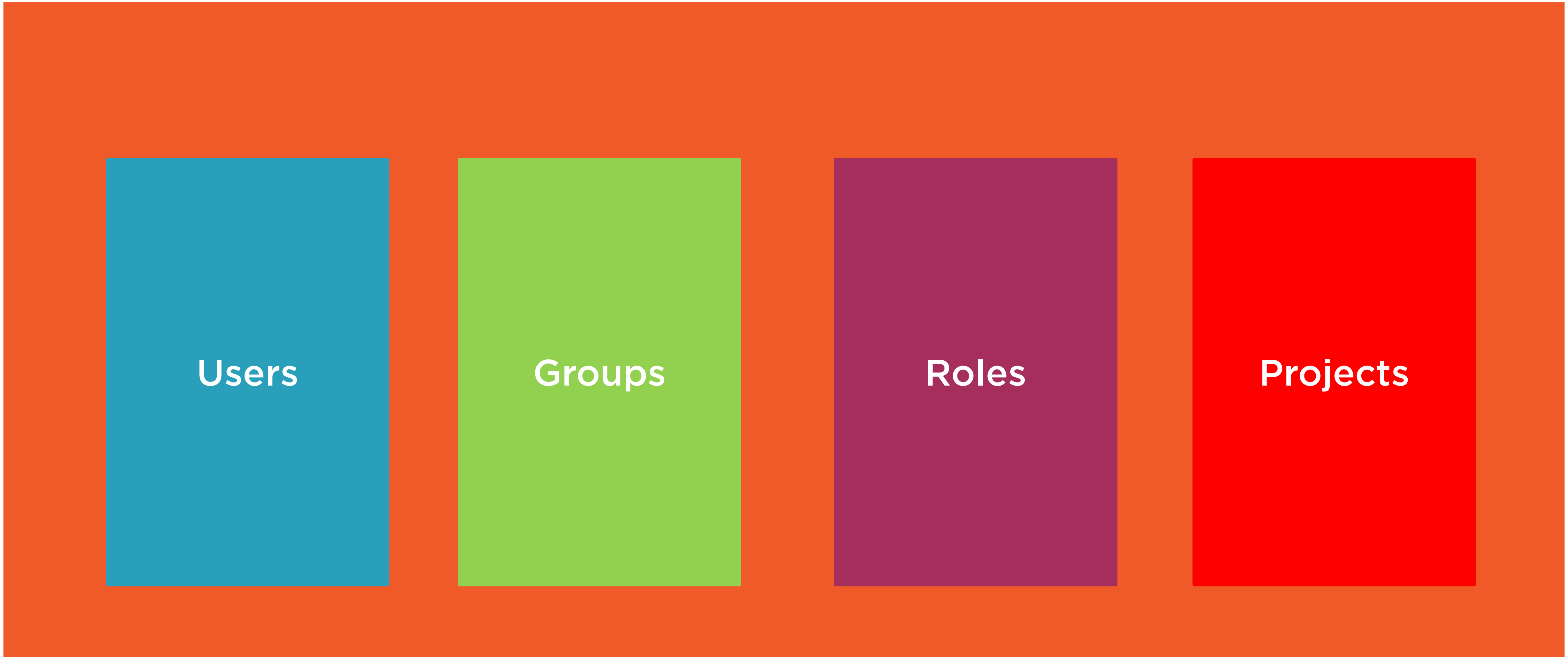
Create Endpoints

When creating service entries for Keystone, port 5000 is the user port and port 35357 is reserved for admin. The Version 3 API is denoted with v3 not v3.0

Domain

Domains are a collection of projects and users that define administrative boundaries for managing Identity entities. Domains can represent an individual, company, or operator-owned space. Users can be granted the administrator (admin) role for a domain. A domain administrator can create projects, users, and groups in a domain and assign roles to users and groups in a domain.

Domain



Users

Groups

Roles

Projects

```
# source keystonerc_adminv3

# openstack domain create pluralsight
```

Create a New Domain

As the main stack admin we can create a new domain

```
# openstack user create --password Password1 \
  --domain pluralsight ps_admin

# openstack user list

# openstack user list --domain pluralsight
```

Add User to New Domain

Here we create a new user called ps_admin in the Pluralsight domain

```
# openstack role add --user ps_admin \
  --domain pluralsight admin

# openstack role list --user ps_admin --domain pluralsight
```

Make User Administrator of Domain

Adding the user to the admin role for the domain will assign them all privileges within the domain. The admin role is automatically defined.

Credentials File

```
unset OS_SERVICE_TOKEN
export OS_USERNAME=ps_admin
export OS_PASSWORD=Password1
export OS_AUTH_URL=http://192.168.56.5:5000/
export OS_IDENTITY_API_VERSION=3
export PS1='[\u@\h \W(ps_admin)]\$ ‘
export OS_USER_DOMAIN_NAME=pluralsight
export OS_DOMAIN_NAME=pluralsight
export OS_REGION_NAME=RegionOne
```

Manage Domains

Projects

A project is a container that groups or isolates resources or identity objects. Projects were previously called tenants and were the main OpenStack containers before Domains. Now we have Domains these are very much more like Organizational Units in LDAP directories creating sub-groupings for administration

```
# openstack project create slc

# openstack project list

# openstack project show slc

# openstack user create --password Password1 leah

# openstack role add --project slc --user leah _member_
```

Projects in OpenStack

Projects allow resources to be grouped. Although not created in the project, users gain privileges via role membership.

Manage Projects

Groups

Another V3 API Identity object that allows users to be gathered together and the group to be added to a role. Like users, groups are created within the domain

```
# openstack group list

# openstack group create --domain pluralsight ops

# openstack group add user ops bob

# openstack group list --user bob
```

OpenStack Groups

Groups are created within domain to manage users and roles effectively

```
OS_AUTH_URL=http://192.168.56.5:5000/
OS_IDENTITY_API_VERSION=3
```

Services

Endpoints

Domains

Users

Roles

Projects

Groups

Next up: Image Management with Glance