

Installing the Keystone Identity Service



Andrew Mallett

LINUX AUTHOR AND TRAINER

@theurbanpenguin www.theurbanpenguin.com



Objectives



Create Database for Keystone

Install Keystone

Configure Apache Vhost

Create Identity Elements



OpenStack Operating System

Keystone Identity Service

Keystone is the project name for OpenStack Identity, a service that provides token, policy, and catalog functions via an OpenStack application programming interface (API).

Keystone itself just provides the mechanism to stores these Identities, in itself it does not store and end User information.

Keystone allows organizations to make use of their own backends and Keystone will interface with OpenStack. We use MySQL as our backend storage



Create the Database

MYSQL_ROOT_PW=Password1

```
cat > create-keystonedb.sql << END
```

```
CREATE DATABASE keystone;
```

```
GRANT ALL PRIVILEGES ON keystone.* TO 'keystone'@'localhost' IDENTIFIED  
BY 'Password1';
```

```
GRANT ALL PRIVILEGES ON keystone.* TO 'keystone'@'%' IDENTIFIED BY  
'Password1';
```

```
SHOW GRANTS FOR 'keystone'@'%'  
END
```

```
mysql -u root -p$MYSQL_ROOT_PW < create-keystonedb.sql
```



Create Database for Keystone



Install Keystone

```
# echo "manual" > /etc/init/keystone.override
```

```
# apt-get install keystone \  
  apache2 \  
  libapache2-mod-wsgi \  
  memcached \  
  python-memcache
```

We don't want keystone to start so we override it first

The web-servlet in keystone is depreciated in favor of
the apache wsgi module.



/etc/keystone/keystone.conf

[DEFAULT]

verbose = True

admin_token = Password1

log_dir = /var/log/keystone

[database]

connection = mysql://keystone:Password1@controller/keystone

[memcache]

servers = localhost:11211

[revoke]

driver = sql

[token]

provider = uuid

driver = memcache

[extra_headers]

Distribution = Ubuntu



```
# su -s /bin/sh -c "keystone-manage db_sync" keystone
```

```
# mysql -u root -p
```

- use keystone;
- show tables;
- quit

Populate Keystone Database

We can create the schema and initial data for the database once the keystone.conf is in place. With the sync complete we can see the schema from within the keystone database




```
# ./create-keystone-apache-vhost.sh
```

Configure Apache VHOST for Keystone

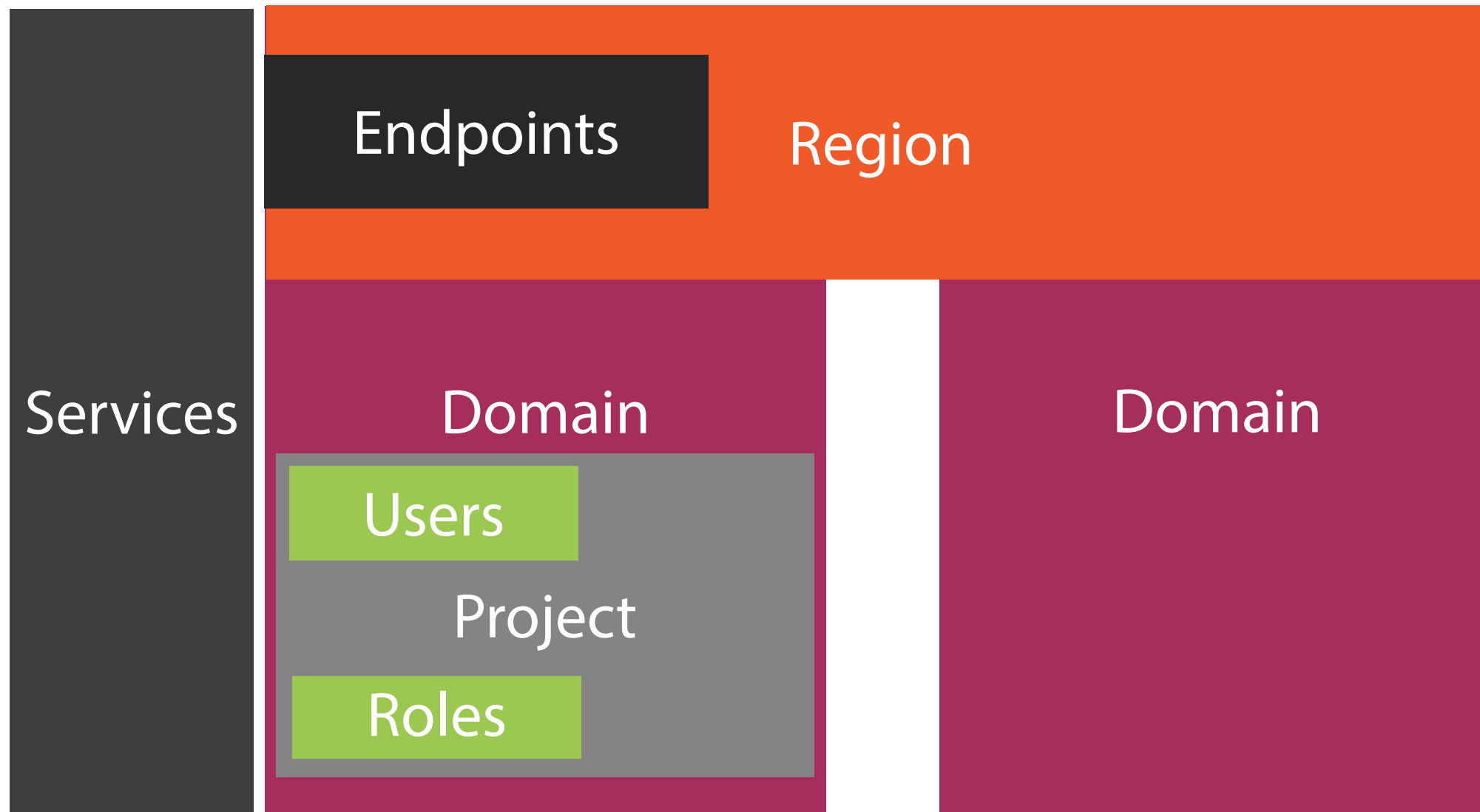
The supplied script will create the vhost for you, along with the keystone configuration. There is a lot of typing in creating these virtual hosts so we omit it from the slide. We also have to set the `ServerName` directive and restart the web server, all of which is managed by the script.



Install Keystone and Create VHOST



Keystone Identities



```
# export OS_TOKEN=Password1  
# export OS_URL=http://controller:35357/v3  
# export OS_IDENTITY_API_VERSION=3
```

Authenticate

We need to authenticate to the administration port of Keystone. As we don't have users yet we can use the token we set in the keystone.conf



```
# openstack service create \  
--name keystone \  
--description "OpenStack Identity" \  
identity  
  
# openstack service list  
  
# openstack service delete keystone
```

Create Identities

Again we can use the scripts to manage this but we will create the top levels entries



```
# openstack endpoint create --region RegionOne identity public  
http://controller:5000/v2.0
```

```
# openstack endpoint create --region RegionOne identity internal  
http://controller:5000/v2.0
```

```
# openstack endpoint create --region RegionOne identity admin  
http://controller:35357/v2.0
```

Endpoints

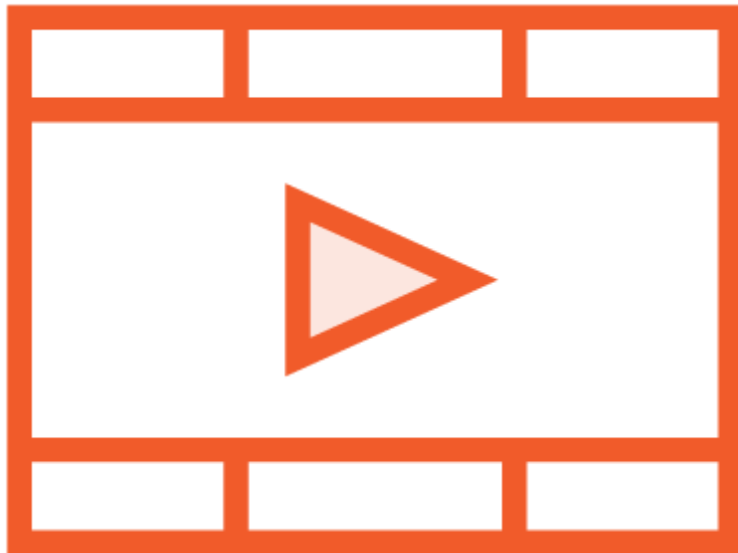
Just a quick note on endpoints. We can have three NICs in OpenStack to separate the traffic: admin, public, and internal

We use the one interface and the IP Address of the Controller



Create Keystone Identities





Created Keystone Database

Installed Keystone and Apache: We use the Apache Service to Interface into Keystone

Configured keystone.conf

Populated Database

Created Keystone Identities: Services, endpoints, projects, users and roles



Next up: Installing the
Glance Image Service

