OpenStack

Identity(keystone)

在controller节点安装和配置认证服务

- 配置先决条件
- 安装并配置认证服务组件
- 完成安装



配置先决条件

- 1、创建认证服务数据库
 - a.登录mysql数据库

#mysql -u root -p

b.创建keystone数据库

CREATE DATABASE keystone;

c.创建keystone数据库用户,使其可以对keystone数据库有完全控制权限

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

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

2、生成一个随机值作为管理令牌在初始配置

openssl rand -hex 10:

更多云计算-Java - 大数据 - 前端 - python人工智能资料下载,可百度访问:尚硅谷官网



安装和配置认证组件

1、安装软件包

yum install openstack-keystone python-keystoneclient

2、编辑/etc/keystone/keyston.conf文件并作下列修改:

a.修改[DEFAULT]小节,定义初始管理令牌。

[DEFAULT]

• • •

admin_token = 刚才生成的随机值

b.修改[database]小节,配置数据库访问

[database]

• • •

connection = mysql://keystone:*KEYSTONE_DBPASS*@controller.nice.com/keystone



安装和配置认证组件

c.修改[token]小节,配置UUID提供者和SQL驱动

```
[token]
```

. . .

provider = keystone.token.providers.uuid.Provider driver = keystone.token.persistence.backends.sql.Token

d. (可选)开启详细日志,协助故障排除

[DEFAULT]

...

verbose = True



安装和配置认证组件

3、常见通用证书的密钥,并限制相关文件的访问权限

```
# keystone-manage pki_setup --keystone-user keystone --keystone-group keystone
# chown -R keystone:keystone /var/log/keystone
# chown -R keystone:keystone /etc/keystone/ssl
# chmod -R o-rwx /etc/keystone/ssl
```

4、初始化keystone数据库

su -s /bin/sh -c "keystone-manage db sync" keystone



完成安装

1、启动identity服务并设置开机启动

```
# systemctl enable openstack-keystone.service
# systemctl start openstack-keystone.service
```

2、默认情况下,服务器会无限存储到期的令牌,在资源有限的情况下会严重影响服务器性能。建议用计划任务,每小时删除过期的令牌

```
# (crontab -I -u keystone 2>&1 | grep -q token_flush) | | \
echo '@hourly /usr/bin/keystone-manage token_flush >/var/log/keystone/
keystone-tokenflush.log 2>&1' \
>> /var/spool/cron/keystone
```



创建tenants(租户),(users)用户和(roles)角色

- 配置先决条件
- 创建租户,用户和角色。



配置先决条件

1、配置管理员令牌

#export OS_SERVICE_TOKEN=例才生成的字符串

2、配置端点

#export OS_SERVICE_ENDPOINT=http://controller.nice.com:35357/v2.0



创建租户,用户和角色

- 1、创建用于管理的租户,用户和角色
 - a.创建admin租户

#keystone tenant-create --name admin --description "Admin Tenant"



b.创建admin用户

#keystone user-create --name admin --pass ADMIN_PASS --email EMAIL_ADDRESS

更多云计算-Java - 大数据 - 前端 - python人工智能资料下载,可百度访问:尚硅谷官网



c.创建admin角色

#keystone role-create --name admin



- d.添加admin租户和用户到admin角色
- # keystone user-role-add --tenant admin --user admin --role admin
- e.创建用于dashboard访问的"_member_"角色

#keystone role-create --name _member_



- f.添加admin租户和用户到_member_角色
- #keystone user-role-add --tenant admin --user admin --role _member_



2、创建一个用于演示的demo租户和用户

a.创建demo租户

#keystone tenant-create --name demo --description "Demo Tenant"



b.创建的demo用户

#keystone user-create --name demo --pass DEMO_PASS --email EMAIL_ADDRESS



- c.添加demo租户和用户到_member_角色
- #keystone user-role-add --tenant demo --user demo --role _member_
- 3、OpenStack服务业需要一个租户,用户和角色和其他服务进行交互。因此我们创建一个service的租户。任何一个OpenStack服务都要和它关联
 - #keystone tenant-create --name service --description "Service Tenant"

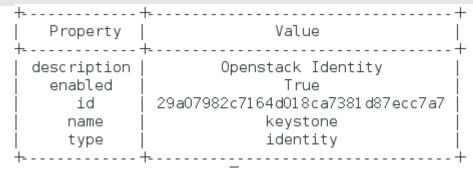


创建服务实体和API端点

1、在OpenStack环境中,identity服务管理一个服务目录,并使用这个目录在OpenStack环境中定位其他服务。

为identity服务创建一个服务实体

#keystone service-create --name keystone --type identity --description "OpenStack Identity"





2、OpenStack环境中,identity服务管理目录以及与服务相关API断点。服务使用这个目录来沟通其他服务。

OpenStack为每个服务提供了三个API端点: admin(管理),internal(内部),public(公共)为identity服务创建API端点

```
#keystone endpoint-create \
--service-id $(keystone service-list | awk '/ identity / {print $2}') \
--publicurl http://controller.nice.com:5000/v2.0 \
--internalurl http://controller.nice.com:5000/v2.0 \
--adminurl http://controller.nice.com:35357/v2.0 \
--region regionOne
```



确认操作

1、删除OS_SERVICE_TOKEN 和OS_SERVICE_ENDPOINT 临时变量

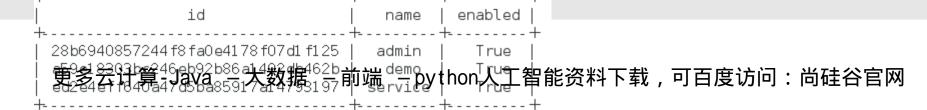
unset OS_SERVICE_TOKEN OS_SERVICE_ENDPOINT

2、使用admin租户和用户请求认证令牌

keystone --os-tenant-name admin --os-username admin --os-password *ADMIN_PASS* --os-auth-url http://controller.nice.com:35357/v2.0 token-get

3、以admin租户和用户的身份查看租户列表

#keystone --os-tenant-name admin --os-username admin --os-password *ADMIN_PASS* --os-auth-url http://controller.nice.com:35357/v2.0 tenant-list





4、以admin租户和用户的身份查看用户列表

keystone --os-tenant-name admin --os-username admin --os-password *ADMIN_PASS* --os-auth-url http://controller.nice.com:35357/v2.0 user-list

5、以admin租户和用户的身份查看角色列表

keystone --os-tenant-name admin --os-username admin --os-password *ADMIN_PASS* --os-auth-url http://controller.nice.com:35357/v2.0 role-list

```
+-----+
| id | name |
+------+
| a680862d8d8f455aa5f9a182894b402d | _member_ |
| d5ec9f92280c4e97af8b4b85a7ec5605 | admin |
+-----+
```



6、以demo租户和用户的身份请求认证令牌

keystone --os-tenant-name demo --os-username demo --os-password *DEMO_PASS* --os-auth-url http://controller.nice.com:35357/v2.0 token-get

7、以demo租户和用户的身份查看用户列表

keystone --os-tenant-name demo --os-username demo --os-password *DEMO_PASS* --os-auth-url http://controller.nice.com:35357/v2.0 user-list

```
You are not authorized to perform the requested action: admin_required (HTTP 403)
```



创建OpenStack客户端环境脚本

为了方便使用上面的环境变量和命令选项,我们为admin和demo租户和用户创建环境脚本。

1、编辑admin-openrc.sh

```
export OS_TENANT_NAME=admin
export OS_USERNAME=admin
export OS_PASSWORD=ADMIN_PASS
export OS_AUTH_URL=http://controller.nice.com:35357/v2.0
```

2、编辑demo-openrc.sh

```
export OS_TENANT_NAME=demo
export OS_USERNAME=demo
export OS_PASSWORD=DEMO_PASS
export OS_AUTH_URL=http://controller.nice.com:5000/v2.0
```

加载客户端环境脚本

#source admin-openrc.sh

更多云计算-Java - 大数据 - 前端 - python人工智能资料下载,可百度访问:尚硅谷官网

