Potek naloge 3

Najprej bomo ustvarili administratorja in stanovalca. Da bi lahko dostopali do storitev, je potrebno registrirati in ustvariti dostopne točke komponent platforme. Za dostop administratorjev, uporabnikov in procesov do komponente ter storitev je potrebno podati avtorizacijske in identifikacijske podatke.

Postopek je povzet po administraciji platforme OpenStack.

Najprej bomo ustvarili identifikacijski žeton, ki se uporablja pri komunikaciji med storitvami. Žeton ustvarimo z ukazom:

$ openssl rand – hex 10

Tolmač ukazov nam vrne naključen 20-mestni niz kombinacij številk in črk:

4d49d6eb54f2aefff673

 Pri tem je potrebno biti pozoren na to, da vsak dobi drugačen žeton.

V tem primeru sem jaz dobil žeton *4d49d6eb54f2aefff673*.

Ustvarjen žeton vnesemo v nastavitveno datoteko */etc/keystone/keystone.conf* , pod razdelek *[DEFAULT]* in nastavimo dnevniško datoteko (*ang. log file*), kamor se vpisujejo dogodki komponente Keystone. Tako se kasneje lažje odkrije napaka.

```
[DEFAULT]
admin_token = 4d49d6eb54f2aefff673
log_dir = /var/log/keystone
```

$sudo service keystone restart

Da lahko dostopamo do storitev Keystona, se moramo avtenticirati preko okoljskih spremenljivk (*ang. environment variables*) *OS_SERVICE_TOKEN* in *OS_SERVICE_ENDPOINT*, ki jih izvozimo z ukazom *export*:

$ export OS_SERVICE_TOKEN = 4d49d6eb54f2aefff673 //uporabimo prej ustvarjeni žeton

$ export OS_SERVICE_ENDPOINT = http://192.168.200.26:35357/v2.0 //nastavimo dostopno //točko, preko katere dostopamo do storitve

V kolikor nas zanimajo vsi možni ukazi komponente, uporabimo *$ <ime_komponente> --help*. Primer:

$ keystone --help

V kolikor nas zanima opis posameznega ukaza, uporabimo $<*ime_komponente*> *help* <*ukaz*>. Primer:

$ keystone help user-list

S spodnjim ukazom ustvarimo administratorja *admin*, ki nas predstavlja v vlogi administratorja platforme in s katerim lahko upravljamo s platformo. Nastavimo si geslo in elektronski naslov:

$ keystone user-create --name=admin --pass=adminpassword
--email=<elektronski_naslov_administratorja>

 Potrebno si je zapomniti prijavne podatke skrbnika sistema.

Preverimo, če smo ustvarili uporabnika:

$ keystone user-list

 V kolikor nismo ustvarili uporabnika *admin*, potem smo uporabili napačne prijavne podatke storitve Keystone.

V vlogi administratorja lahko ustvarjamo stanovalce, uporabnike, storitve in vloge. Stanovalci so organizirane enote, ki jim dodamo uporabnike in storitve. Določajo navidezne vire uporabnikov, kot so: število navideznih diskov, število uporabljanih virtualnih strojev in

dodeljevanje naslovov IP. Vloge definirajo, katere ukaze lahko uporabnik izvaja. Ustvarimo vlogo administratorja, ki ima najvišje pravice dostopov in upravljanja oblaka.

$ keystone role-create --name=admin

Ustvarimo stanovalca, kamor bomo dodali administratorja.

$ keystone tenant-create --name=admin --description="Stanovalec Admin"

Povežemo administratorja, stanovalca in dodeljeno vlogo:

$ keystone user-role-add --user=admin --tenant=admin --role=admin

Na koncu je potrebno povezati še administratorja, stanovalca, privzeto vlogo _member_:

*$ keystone user-role-add --user=admin --tenant=admin --role=_member_*

Preverimo, če smo pravilno povezali uporabnika in stanovalca:

$ keystone user-role-list --user admin --tenant admin

Identifikacijska storitev oblaka Keystone mora imeti nadzor nad storitvami, določiti mora njihove dostopne točke in poznati njihove dostopne točke v omrežju. Zato je vsako storitev potrebno ustvariti kot uporabnika, ustvariti storitev, dodeliti stanovalca in dostopne točke.
Ker stanovalca še nismo ustvarili, bomo začeli z njim:

$ keystone tenant-create --name=service --description=»Stanovalec procesi platforme«

Vse storitve bodo uporabljale istega stanovalca *service*. Tako bodo imele zagotovljen dostop do ostalih storitev z najvišjimi dostopnimi pravicami. Storitev ustvarimo podobno kot uporabnika. Nato ustvarimo storitev:

$ keystone service-create --name=keystone --type=identity
--description="OpenStack storitev identifikacije"

Tolmač ukazov nam izpiše registrirano storitev:

```
+-------------+----------------------------------+
| Property    |              Value               |
+-------------+----------------------------------+
| description | OpenStack proces idetifikacije   |
| id          | 858e325fd29f4de6949bb1c6c66fe5c1 |
| name        | keystone                         |
| type        | identity                         |
+-------------+----------------------------------+
```

Kot vidimo, ima ustvarjena storitev svoj ID, preko katerega ga identifikacijska storitev prepozna, ID storitve vnesemo pri ustvarjanju dostopne točke. Dostopna točka poveže API komponente s storitvijo. Vsaka storitev ima lastno dostopno točko. Le-ta definira vrata za dostope preko zunanjega in notranjega omrežja. Nastavimo vrata, do katerih ima dostop administrator platforme. Dostop do storitve iz zunanjega in notranjega omrežja je na vratih 5000. Administrator ima dostop do storitev preko vrat 35357.

```
$ keystone endpoint-create  --service-id= 858e325fd29f4de6949bb1c6c66fe5c1
 --publicurl=http:// <ime_našegastrežnika>:5000/v2.0
--internalurl=http:// <ime_našegastrežnika>:5000/v2.0
--adminurl=http:// <ime_našegastrežnika>:35357/v2.0
```

Nato nam tolmač ukazov izpiše pravkar ustvarjeno dostopno točko:

```
+-------------+----------------------------------+
|  Property   |              Value               |
+-------------+----------------------------------+
| adminurl    | http://<ime_našegastrežnika>:35357/v2.0 |
| id          | 11f9c625a3b94a3f8e66bf4e5de2679f |
| internalurl | http:// <ime_našegastrežnika>:5000/v2.0 |
| publicurl   | http:// <ime_našegastrežnika>:5000/v2.0 |
| region      | regionOne                        |
| service_id  | 15c11a23667e427e91bc31335b45f4bd |
+-------------+----------------------------------+
```

Ker nismo imeli ustvarjene dostopne točke, smo do storitev morali dostopati z okoljskimi spremenljivkami. Poleg okoljskih spremenljivk lahko do vseh storitev platforme dostopamo s prijavnimi parametri v ukazu. Najprej je potrebo onemogočiti uporabo okoljskih spremenljivk:

$ unset OS_SERVICE_TOKEN OS_SERVICE_ENDPOINT

Prijavne podatke navedemo v ukazu:

$ keystone --os-username=admin --os-password=adminpassword --os-auth-url=http://<ime_našegastrežnika>:35357/v2.0 token-get

Nazaj dobimo žeton, ki je povezan z administratorjevimi prijavnimi podatki. S tem preverimo, če se storitev nahaja na nastavljeni dostopni točki in če administratorski račun vsebuje navedene prijavne podatke.

 V primeru napake nas obvesti, da manjkajo podatki za avtentikacijo.

Razlika med avtentikacijo in avtorizacijo je, da avtentikacija preverja identiteto uporabnika (uporabniška gesla), avtorizacija pa izvaja nadzor dostopa do določenih storitev.

S spodnjim ukazom preverimo, če se avtorizacija pravilno izvede:

$ keystone --os-username=admin --os-password=adminpasss --os-tenant-name=admin --os-auth-url=http://<ime_našega_strežnika>:35357/v2.0 token-get

Nazaj sprejmemo žeton.

 Pri tem vsak sprejme svoj žeton.

61892fda89ea4bf889708a2a452bd77

Žeton vsebuje ID stanovalca, na katerega je povezan administrator. To pomeni, da je uporabniški račun povezan s točno določenim stanovalcem.

Namesto uporabe okoljskih spremenljivk lahko ustvarimo datoteko *admin.sh*, v katero vpišemo prijavne podatke za administratorja, ki smo jih prej ustvarili, in jo vsakič izvozimo pri delu s storitvami:

```
$ sudo nano admin.sh
export OS_USERNAME=admin                    //nastavimo ime uporabnika
export OS_PASSWORD=adminpassword            //nastavimo geslo uporabnika
export OS_TENANT_NAME=admin                 //nastavimo ime stanovalca
export OS_AUTH_URL=http://<ime_našega_strežnika>:353578/v2.0   //nastavimo
naslov, preko //katerega se bo izvajala avtorizacija
$ source admin.sh
```

Nato preverimo, če smo pravilno napisali podatke in, če se izvede avtorizacija:

```
$ keystone token-get
```

Tolmač ukazov nam vrne podatke o žetonu. V kolikor je ID stanovalca vrnjenega žetona enak našemu (ID stanovalca: *439ac9819f774474a35a36e69945e7c1*), potem smo pravilno nastavili prijavne podatke v datoteki.

Po uspešni registracij storitve Keystone bomo ustvarili in registrirali storitev Glance. Ne bomo podajali natančnega opisa, saj je le-ta podan pri primeru storitve Keystone.

Ustvarimo uporabnika *glance* z geslom *glancepassword* in mu dodamo stanovalca *service* vlogo *admin*. S tem se storitev Glance lahko avtenticira:

```
$ keystone user-create --name=glance --pass=glancepassword
  --email=glance@tecaj.com
$ keystone user-role-add --user=glance --tenant=service
```

--role=admin

 Zapomniti si je potrebno prijavne podatke storitve Glance.

Ustvarimo storitev:

$ keystone service-create --name=glance --type=image
--description="OpenStack storitev slikovne hrambe"

Tolmač ukazov nam vrne ID storitev *(78488899358759996844)* in jo uporabimo pri ustvarjanju dostopne točke. Storitev Glance
uporablja vrata 9292 za vse tri vrste dostopov.

$ keystone endpoint-create
--service-id=78488899358759996844
--publicurl=http://<ime_našega_strežnika>:9292
--internalurl=http://<ime_našega_strežnika>:9292
--adminurl=http://<ime_našega_strežnika>:9292

Nato ustvarimo uporabnika *Nova z* geslom *novapassword* in dodelimo stanovalca *service* ter vlogo *admin*:

$ keystone user-create --name=nova --pass=novapassword
--email=nova@tecaj.com
$ keystone user-role-add --user=nova --tenant=service --role=admin

 Zapomniti si je potrebno prijavne podatke storitve Nova.

Ustvarimo storitev:

$ keystone service-create --name=nova --type=compute
--description="OpenStack racunska storitev"

Tolmač ukazov nam vrne ID storitve *(788f95f599540629859efc3af242a3d)* in ustvarimo njeno dostopno točko, pri kateri na koncu dodamo */%\(tenant_id\)s*, kar poleg administratorskega stanovalca predstavlja tudi stanovalce uporabnikov, ki upravljajo s svojimi navideznimi stroji. Storitev uporablja vrata 8774:

```
$ keystone endpoint-create
  --service-id=788f95f599540629859efc3af242a3d
  --publicurl=http://<ime_našega_strežnika>:8774/v2/%\(tenant_id\)s
  --internalurl=http://<ime_našega_strežnika>8774/v2/%\(tenant_id\)s
  --adminurl=http://<ime_našega_stežnika>:8774/v2/%\(tenant_id\)s
```

Ustvarimo še uporabnika *cinder* z geslom *cinderpass*, ki ga uporablja Cinder za avtentikacijo. Dodelimo mu stanovalca *service* in vlogo *admin*:

```
  $ keystone user-create --name=cinder --pass=cinderpass --
email=cinder@tecaj.com
  $ keystone user-role-add --user=cinder --tenant=service --role=admin
```

 Zapomniti si je potrebno prijavne podatke storitve Cinder.

Ustvariti je potrebno proces, ki ga bodo uporabljale ostale storitve platforme, in narediti je potrebno dostopno točko:

```
$ keystone service-create --name=cinder --type=volume --description="OpenStack
blokovna shramba"
```

Ponovno sprejmemo ID storitve (a*57bf91637bd4645893e40b753dd6a3f*) in ustvarimo dostopno točko za storitve, ki bodo uporabljale storitev Glance. Storitev uporablja vrata 8776. Tako kot pri storitvi Nova moramo na koncu dodati *(tenant_id\)s*, ki omogoča dostop do vseh stanovalcev, saj tako lahko uporabniki ustvarjajo svoje blokovne naprave in jih pripnejo virtualnim strojem. Začnemo z dostopno točko:

$ keystone endpoint-create
--service-id= a57bf91637bd4645893e40b753dd6a3f
--publicurl=http://<ime_našega_strežnika>8776/v1/%\(tenant_id\)s
--internalurl=http://<ime_našega_strežnika>:8776/v1/%\(tenant_id\)s
--adminurl=http://<ime_našega_strežnika>:8776/v1/%\(tenant_id\)s

Ustvariti je potrebno še storitev *cinderv2*, ki se uporablja pri dostopu administratorjev in uporabnikov do Glancovega API-ja:

$ keystone service-create --name=cinderv2 --type=volumev2 --
description="OpenStack blokovna shramba 2 "

Ustvarimo še dostopno točko:

$ keystone endpoint-create
--service-id= b9277b6d355a418aa9e888a50a0409e1
--publicurl=http://<ime_našega_strežnika>:8776/v2/%\(tenant_id\)s
--internalurl=http://<ime_našega_strežnika>:8776/v2/%\(tenant_id\)s
--adminurl=http://<ime_našega_strežnika>:8776/v2/%\(tenant_id\)s

Na koncu bomo ustvarili navadnega uporabnika z imenom *user* in geslo *userpassword*, ki ga bomo uporabili v naslednji nalogi.

$ keystone user-create --name=user --pass=userpassword
--email=<elektronski_naslov_uporbnika>

 Zapomniti si je potrebno prijavne podatke uporabnika *user*.

Ustvarimo stanovalca *user*, kamor bomo dodali navadnega uporabnika.

$ keystone tenant-create --name=user --description="Stanovalec Uporabnik"

Potem povežemo uporabnika *user*, privzeto vlogo _member_: in stanovalca *user*

*$ keystone user-role-add --user=user --role=_member_*

--tenant=user

Prijavne podatke shranimo v datoteko *uporabnik.sh*:

$ sudo nano uporabnik.sh
export OS_USERNAME=user
export OS_PASSWORD=userpassword
export OS_TENANT_NAME=user
export OS_AUTH_URL=http://<ime_našega_strežnika>:353578/v2.0