Table of Contents

Installazione e configurazione del servizio Identity (Keystone) **1**

Installazione Keystone **2**

Configurazione keystone **3**

Creazione di tenant, utente e ruolo per l'amministratore e del tenant per i servizi **4**
- Creazione del tenant "adminTenant" 4
  - Creazione dell'utente "adminUser" 4
  - Creazione dei ruoli admin e memberRole 4
  - Assegnazione del ruolo "admin" all'utente "AdminUser" 5
  - Creazione del tenant "service": 5

Creazione ed inserimento degli utenti associati ai servizi **6**
- Creazione ed inserimento dell'utente associato a Glance 6
  - Creazione ed inserimento dell'utente associato a Nova 6
  - Creazione ed inserimento dell'utente associato a EC2 7
  - Creazione ed inserimento dell'utente associato a Swift 7

Definizione dei servizi **8**
- Definizione del servizio identity (Keystone) 8
  - Definizione del servizio compute (Nova) 8
  - Definizione del servizio volume (Volume) 9
  - Definizione del servizio image (Glance) 9
  - Definizione del servizio compatibility (EC2) 10
  - Definizione del servizio object storage (Swift) 10

Troubleshooting **12**

Installazione e configurazione del servizio Identity (Keystone)

Installazione Keystone

Server designato per l'installazione: hostname1.domain.

- Installare Keystone tramite il gestore di pacchetti `yum` :

```
# yum install openstack-utils openstack-keystone python-keystoneclient
```

- Installare il server `mysql`:

```
# yum install mysql mysql-server MySQL-python
```

- Permettere che il servizio `mysqld` venga avviato di default al riavvio del server ed avviare il servizio:

```
# chkconfig mysqld on
# service mysqld start
```

- Inserire la password di `root` :

```
# mysqladmin -u root password *******
```

Configurazione keystone

- Rimuovere, se esiste, il file `/var/lib/keystone/keystone.db` (cioè il DB sqlite). Verrà utilizzato MySQL che permette a più Keystone di utilizzare lo stesso DB e al DB stesso di essere replicato utilizzando le feature di MySQL.

- Per creare il DB keystone e l'utente keystone che ha completo accesso al DB stesso, digitare il comando:

```
# openstack-db --init --service keystone
```

- Controllare in `/etc/keystone/keystone.conf` che esista una riga del tipo:

```
connection = mysql://keystone:keystone@hostname1.domain/keystone
```

  Nota bene: in `keystone:keystone` il primo keystone è il nome utente, il secondo è la password. Utilizzando il comando `openstack-db --init --service keystone` per l'inizializzazione del DB keystone verranno utilizzati quei valori di default.

- Inizializzare il parametro `admin_token` contenuto all'interno del file `/etc/keystone/keystone.conf` , attraverso il comando:

```
# openstack-config --set /etc/keystone/keystone.conf DEFAULT admin_token $(openssl rand -h
```

- Assicurarsi che il servizio sia attivo al boot ed riavviarlo attraverso i seguenti comandi:

```
# chkconfig openstack-keystone on
# service openstack-keystone restart
```

- Infine inizializzare il DB keystone:

```
# keystone-manage db_sync
```

Creazione di tenant, utente e ruolo per l'amministratore e del tenant per i servizi

Per comodità in questa sezione si utilizzeranno delle variabili d'ambiente settate come segue:

```
# export ADMIN_TOKEN=<ADMIN_TOKEN_VALUE>
export OS_USERNAME=adminUser
export OS_PASSWORD=<PASSWORD>
export OS_TENANT_NAME=adminTenant
export ENDPOINT=http://openstack-01.cnaf.infn.it:35357/v2.0/
export OS_AUTH_URL=http://openstack-01.cnaf.infn.it:5000/v2.0/
```

Dove <ADMIN_TOKEN_VALUE> è il valore del parametro `admin_token` contenuto nel file `/etc/keystone/keystone.conf`.

Creazione del tenant "adminTenant"

Definizione di tenant: *Un tenant è un contenitore usato per raggruppare risorse o utenti.*

- Creare il tenant lanciando il comando:

```
keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT tenant-create --name adminTenant --desc
+-------------+----------------------------------+
|   Property  |              Value               |
+-------------+----------------------------------+
| description | Admin Tenant                     |
| enabled     | True                             |
| id          | db2cf825309c49989595fc2ff915dc7e |
| name        | adminTenant                      |
+-------------+----------------------------------+

export ADMIN_TENANT_ID=db2cf825309c49989595fc2ff915dc7e
```

Creazione dell'utente "adminUser"

- Creare l'utente lanciando il seguente comando:

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT user-create --tenant_id $ADMIN_TENANT_ID --n
+----------+-------------------------------------------------------------------------------------
| Property |                                                                              Value
+----------+-------------------------------------------------------------------------------------
| email    | None
| enabled  | True
| id       | 7d6a3a42c37948b88e2fa692b63587cd
| name     | adminUser
| password | $6$rounds=40000$NxZyxUfO8VRj3gR.$zt9GJKwMDOUMDHCMhqAqJje3JAJmqqTXADZkXll.usGHEsEpAMg
| tenantId | db2cf825309c49989595fc2ff915dc7e
+----------+-------------------------------------------------------------------------------------
```

Dove <ADMIN_TENANT_ID> è l'ID del tenant appena creato, mentre <ADMIN_PASSWORD> è la password scelta per l'utente "adminUser".

Creazione dei ruoli admin e memberRole

- Creare il ruolo lanciando il seguente comando:

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT role-create --name admin
```

```
+----------+----------------------------------+
| Property |              Value               |
+----------+----------------------------------+
| id       | 2f196d11ff954c67befc3f190195f47c |
| name     | admin                            |
+----------+----------------------------------+

# export ADMIN_ROLE_ID=2f196d11ff954c67befc3f190195f47c

# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT role-create --name memberRole
+----------+----------------------------------+
| Property |              Value               |
+----------+----------------------------------+
| id       | d193b58e977d4c398b41c518f3629ea7 |
| name     | memberRole                       |
+----------+----------------------------------+
```

Assegnazione del ruolo "admin" all'utente "AdminUser"

- Assegnare il ruolo tramite il seguente comando:

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT user-role-add --user $ADMIN_USER_ID --
```

Dove <ADMIN_USER_ID> e <ADMIN_TENANT_ID> sono rispettivamente gli ID di user e tenant appena creati. $ADMIN_ROLE_ID è invece l'id del ruolo "admin" esportato nell'istruzione precedente. **Nota bene:** non viene visualizzato nulla se il comando ha successo.

Creazione del tenant "service":

Questo tenant conterrà tutti gli utenti dei servizi che si desidera far conoscere al service catalog.

- Creare il tenant lanciando il comando:

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT tenant-create --name service --descrip
+-------------+----------------------------------+
|   Property  |              Value               |
+-------------+----------------------------------+
| description | Service Tenant                   |
| enabled     | True                             |
| id          | 73016aa2c9ca4aeba3736cf44cc8433b |
| name        | service                          |
+-------------+----------------------------------+

# export SERVICE_TENANT_ID=73016aa2c9ca4aeba3736cf44cc8433b
```

Creazione ed inserimento degli utenti associati ai servizi

Una volta creato il "Service Tenant", si andrà ad inserire al suo interno un utente per ogni servizio che si vuole aggiungere.

Creazione ed inserimento dell'utente associato a Glance

- Creare l'utente tramite il seguente comando:

```
keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT user-create --tenant_id $SERVICE_TENANT_ID --n
+----------+-----------------------------------------------------------------------------------------
| Property |                                                                Value
+----------+-----------------------------------------------------------------------------------------
| email    | None
| enabled  | True
| id       | 90ee5ac051eb4d1eaa543243987968a6
| name     | glance
| password | $6$rounds=40000$LApm.pXGC43cDMhN$J1mDpXad5r2YYNoMKK/P5t0VwXQidKauP/oHaVH5Nm9E7zGQLYa
| tenantId | 73016aa2c9ca4aeba3736cf44cc8433b
+----------+-----------------------------------------------------------------------------------------

export GLANCE_USER_ID=90ee5ac051eb4d1eaa543243987968a6
```

Dove <GLANCE_PASSWORD> è la password che si desidera associare all'utente del servizio Glance.

- Assegnare il ruolo tramite il seguente comando:

```
keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT user-role-add --user $GLANCE_USER_ID --
```

Nota bene: non viene visualizzato nulla se il comando ha successo.

Creazione ed inserimento dell'utente associato a Nova

- Creare l'utente tramite il seguente comando:

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT user-create --tenant_id $SERVICE_TENA
+----------+-----------------------------------------------------------------------------------
| Property |                                                                Value
+----------+-----------------------------------------------------------------------------------
| email    | None
| enabled  | True
| id       | 9b2d55e2f8164be5a9805a39588f4659
| name     | nova
| password | $6$rounds=40000$jAXTAHvAnF4MBz7O$pr99MMc4gpfOOlYDO7dTTCy7Ai.XZ72P1GbMXHpj1ri7
| tenantId | 73016aa2c9ca4aeba3736cf44cc8433b
+----------+-----------------------------------------------------------------------------------

export NOVA_USER_ID=9b2d55e2f8164be5a9805a39588f4659
```

Dove <NOVA_PASSWORD> è la password che si desidera associare all'utente del servizio Nova.

- Assegnare il ruolo tramite il seguente comando:

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT user-role-add --user $NOVA_USER_ID --tenant_
```

Nota bene: non viene visualizzato nulla se il comando ha successo.

Creazione ed inserimento dell'utente associato a EC2

- Creare l'utente tramite il seguente comando:

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT user-create --tenant_id $SERVICE_TENA
+----------+-----------------------------------------------------------------------------
| Property |                                                                      Value
+----------+-----------------------------------------------------------------------------
| email    | None
| enabled  | True
| id       | ea9ed6847b814e8e8f315b15ac987ce3
| name     | ec2
| password | $6$rounds=40000$egKz.H1MiHWIvq9o$wXiMiSx8FuIy2KOltS875ldjv0Q591y3xlEYvWl6Uk.n
| tenantId | 73016aa2c9ca4aeba3736cf44cc8433b
+----------+-----------------------------------------------------------------------------

export EC2_USER_ID=ea9ed6847b814e8e8f315b15ac987ce3
```

  Dove <EC2_PASSWORD> è la password che si desidera associare all'utente del servizio EC2.

- Assegnare il ruolo tramite il seguente comando:

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT user-role-add --user $EC2_USER_ID --t
```

Nota bene: non viene visualizzato nulla se il comando ha successo.

Creazione ed inserimento dell'utente associato a Swift

- Creare l'utente tramite il seguente comando:

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT user-create --tenant_id $SERVICE_TENA
+----------+-----------------------------------------------------------------------------
| Property |                                                                      Value
+----------+-----------------------------------------------------------------------------
| email    | None
| enabled  | True
| id       | 995ee845a29147b7a93443054d185c3b
| name     | swift
| password | $6$rounds=40000$CRLmdT7rWkiLClCL$nZ54lQSF5J.wxUrCgZ37wYPywyx9WqChnAJr3g.XPG.S
| tenantId | 73016aa2c9ca4aeba3736cf44cc8433b
+----------+-----------------------------------------------------------------------------

export SWIFT_USER_ID=995ee845a29147b7a93443054d185c3b
```

  Dove <SWIFT_PASSWORD> è la password che si desidera associare all'utente del servizio Swift.

- Assegnare il ruolo tramite il seguente comando:

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT user-role-add --user $SWIFT_USER_ID --
```

Nota bene: non viene visualizzato nulla se il comando ha successo.

Definizione dei servizi

Keystone agisce anche come catalogo dei servizi per permettere alle altre componenti OpenStack di conoscere quali sono gli endpoint dei vari servizi OpenStack.

- Verificare che il file `/etc/keystone/keystone.conf` contenga la seguente riga:

```
[catalog]
driver = keystone.catalog.backends.sql.Catalog
```

Definizione del servizio identity (Keystone)

- Creare il servizio "keystone":

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT service-create --name=keystone --type=
+-------------+----------------------------------+
|   Property  |              Value               |
+-------------+----------------------------------+
| description | Keystone Identity Service        |
| id          | c2b7f0514dde412ea6c1fccac6437bb7 |
| name        | keystone                         |
| type        | identity                         |
+-------------+----------------------------------+

export KEYSTONE_SERVICE_ID=c2b7f0514dde412ea6c1fccac6437bb7
export KEYSTONE5000=http://openstack-01.cnaf.infn.it:5000/v2.0
```

- Creare l'endpoint per il servizio "keystone":

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT endpoint-create --region RegionOne --
+-------------+-----------------------------------------------+
|   Property  |                     Value                     |
+-------------+-----------------------------------------------+
| adminurl    | http://openstack-01.cnaf.infn.it:35357/v2.0/ |
| id          | 93c84640b7384911afdd27dda19ea69c              |
| internalurl | http://openstack-01.cnaf.infn.it:5000/v2.0    |
| publicurl   | http://openstack-01.cnaf.infn.it:5000/v2.0    |
| region      | RegionOne                                     |
| service_id  | c2b7f0514dde412ea6c1fccac6437bb7              |
+-------------+-----------------------------------------------+
```

Definizione del servizio compute (Nova)

Il servizio compute richiede un endpoint specifico per ogni tenant. La stringa `%(tenant_id)s` ed i singoli apici che racchiudono i valori di `publicurl`, `internalurl` e `adminurl` devono essere digitati esattamente come mostrato sia per l'endpoint **compute** che per l'endpoint **volume** e l'endpoint **object storage** (paragrafi successivi).

- Creare il servizio "nova":

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT service-create --name=nova --type=comp
+-------------+----------------------------------+
|   Property  |              Value               |
+-------------+----------------------------------+
| description | Nova Compute Service             |
| id          | 5bfe94c4ff80410ab60b635cc99e2476 |
| name        | nova                             |
| type        | compute                          |
+-------------+----------------------------------+
```

```
export NOVA_COMPUTE_SERVICE_ID=5bfe94c4ff80410ab60b635cc99e2476
```

- Creare l'endpoint per il servizio "nova":

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT endpoint-create --region RegionOne --
  --internalurl='http://<CLOUD_CONTROLLER_HOSTNAME>:8774/v2/%(tenant_id)s' --adminurl='http
+-------------+---------------------------------------------------------+
|   Property  |                          Value                          |
+-------------+---------------------------------------------------------+
| adminurl    | http://openstack-01.cnaf.infn.it:8774/v2/%(tenant_id)s |
| id          | 6168989f6fd2429d8f15d62c940a2fc2                        |
| internalurl | http://openstack-01.cnaf.infn.it:8774/v2/%(tenant_id)s |
| publicurl   | http://openstack-01.cnaf.infn.it:8774/v2/%(tenant_id)s |
| region      | RegionOne                                               |
| service_id  | 5bfe94c4ff80410ab60b635cc99e2476                        |
+-------------+---------------------------------------------------------+
```

Dove <CLOUD_CONTROLLER_HOSTNAME> è l'hostname del Cloud Controller (Nova). Nel nostro caso è "openstack-01.cnaf.infn.it".

Definizione del servizio volume (Volume)

Il servizio volume richiede un endpoint specifico per ogni tenant.

- Creare il servizio "volume":

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT service-create --name=volume --type=v
+-------------+----------------------------------+
|   Property  |              Value               |
+-------------+----------------------------------+
| description | Nova Volume Service              |
| id          | f99ba2241e014295aa3ecfcda6633100 |
| name        | volume                           |
| type        | volume                           |
+-------------+----------------------------------+

export NOVA_VOLUME_SERVICE_ID=f99ba2241e014295aa3ecfcda6633100
```

- Creare l'endpoint per il servizio "volume":

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT endpoint-create --region RegionOne --
+-------------+---------------------------------------------------------+
|   Property  |                          Value                          |
+-------------+---------------------------------------------------------+
| adminurl    | http://openstack-01.cnaf.infn.it:8776/v1/%(tenant_id)s |
| id          | 78b405450a5d432b8a4c61bc1abc52d9                        |
| internalurl | http://openstack-01.cnaf.infn.it:8776/v1/%(tenant_id)s |
| publicurl   | http://openstack-01.cnaf.infn.it:8776/v1/%(tenant_id)s |
| region      | RegionOne                                               |
| service_id  | f99ba2241e014295aa3ecfcda6633100                        |
+-------------+---------------------------------------------------------+
```

Dove <CLOUD_CONTROLLER_HOSTNAME> è l'hostname del Cloud Controller (Nova). Nel nostro caso è "openstack-01.cnaf.infn.it".

Definizione del servizio image (Glance)

- Creare il servizio "glance":

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT service-create --name=glance --type=i
+-------------+----------------------------------+
```

```
|   Property  |              Value               |
+-------------+----------------------------------+
| description | Glance Image Service             |
| id          | 6c02494e7a534c91a7a01228a4093e63 |
| name        | glance                           |
| type        | image                            |
+-------------+----------------------------------+

export GLANCE_IMAGE_SERVICE_ID=6c02494e7a534c91a7a01228a4093e63
```

- Creare l'endpoint per il servizio "glance":

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT endpoint-create --region RegionOne --
+-------------+-----------------------------------------+
|   Property  |                Value                    |
+-------------+-----------------------------------------+
| adminurl    | http://openstack-01.cnaf.infn.it:9292/v1 |
| id          | 2251cae28faf450a8f86292a237e4fcf         |
| internalurl | http://openstack-01.cnaf.infn.it:9292/v1 |
| publicurl   | http://openstack-01.cnaf.infn.it:9292/v1 |
| region      | RegionOne                                |
| service_id  | 6c02494e7a534c91a7a01228a4093e63         |
+-------------+-----------------------------------------+
   Dove &lt;GLANCE_SERVER_HOSTNAME&gt; è l'hostname del server sui è installato il servizi
```

Definizione del servizio compatibility (EC2)

- Creare il servizio "ec2":

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT service-create --name=ec2 --type=ec2
+-------------+----------------------------------+
|   Property  |              Value               |
+-------------+----------------------------------+
| description | EC2 Compatibility Service        |
| id          | 0f30dac463d242668e999955f7ee3d61 |
| name        | ec2                              |
| type        | ec2                              |
+-------------+----------------------------------+

export EC2_COMPATIBILITY_SERVICE_ID=0f30dac463d242668e999955f7ee3d61
```

- Creare l'endpoint per il servizio "ec2":

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT endpoint-create --region RegionOne --
+-------------+---------------------------------------------------------+
|   Property  |                          Value                          |
+-------------+---------------------------------------------------------+
| adminurl    | http://openstack-01.cnaf.infn.it:8773/services/Cloud |
| id          | e927ee99f3b84894a11f9216118d677a                        |
| internalurl | http://openstack-01.cnaf.infn.it:8773/services/Cloud |
| publicurl   | http://openstack-01.cnaf.infn.it:8773/services/Cloud |
| region      | RegionOne                                               |
| service_id  | 0f30dac463d242668e999955f7ee3d61                        |
+-------------+---------------------------------------------------------+
```

Dove <EC2_SERVER_HOSTNAME> è l'hostname del server sui è installato il servizio EC2. **Nel caso di questo prototipo EC2 non è stato installato**.

Definizione del servizio object storage (Swift)

Il servizio object storage richiede un endpoint specifico per ogni tenant.

- Creare il servizio "swift":

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT service-create --name=swift --type=ob
+-------------+----------------------------------+
|   Property  |              Value               |
+-------------+----------------------------------+
| description | Object Storage Service           |
| id          | 5b978ae3aeb1456a9d14c76f1d0c7956 |
| name        | swift                            |
| type        | object-store                     |
+-------------+----------------------------------+

export OBJECT_STORAGE_SERVICE_ID=5b978ae3aeb1456a9d14c76f1d0c7956
```

- Creare l'endpoint per il servizio "swift":

```
# keystone --token $ADMIN_TOKEN --endpoint $ENDPOINT endpoint-create --region RegionOne --
+-------------+-----------------------------------------------------------------+
|   Property  |                              Value                              |
+-------------+-----------------------------------------------------------------+
| adminurl    | http://openstack-04.cnaf.infn.it:8080/v1/AUTH_%(tenant_id)s |
| id          | ef01e39c5dde4757834aba9070132f34                                |
| internalurl | http://openstack-04.cnaf.infn.it:8080/v1/AUTH_%(tenant_id)s |
| publicurl   | http://openstack-04.cnaf.infn.it:8080/v1/AUTH_%(tenant_id)s |
| region      | RegionOne                                                       |
| service_id  | 5b978ae3aeb1456a9d14c76f1d0c7956                                |
+-------------+-----------------------------------------------------------------+
```

Dove <SWIFT_SERVER_HOSTNAME> è l'hostname del server sui è installato il servizio Swift. Nel nostro caso è "openstack-04.cnaf.infn.it".

Troubleshooting

- Per le principali operazioni di troubleshooting riferirsi alla guida .

- Installare curl tramite il gestore di pacchetti `yum` e lanciare il seguente comando per listare le informazioni inserite in Keystone:

```
# yum install curl
# curl -d '{"auth": {"tenantName": "adminTenant", "passwordCredentials":{"username": "admi
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
106  2446  101  2446    0   120  27990   1373 --:--:-- --:--:-- --:--:-- 27046
{
    "access": {
        "serviceCatalog": [
            {
                "endpoints": [
                    {
                        "adminURL": "http://hostname1.domain:8774/v2/c8854bd3d15f4f479476d
                        "internalURL": "http://hostname1.domain:8774/v2/c8854bd3d15f4f4794
                        "publicURL": "http://hostname1.domain:8774/v2/c8854bd3d15f4f479476
                        "region": "RegionOne"
                    }
                ],
                "endpoints_links": [],
                "name": "nova",
                "type": "compute"
            },
            {
                "endpoints": [
                    {
                        "adminURL": "http://hostname1.domain:9292/v1/",
                        "internalURL": "http://hostname1.domain:9292/v1/",
                        "publicURL": "http://hostname1.domain:9292/v1/",
                        "region": "RegionOne"
                    }
                ],
                "endpoints_links": [],
                "name": "glance",
                "type": "image"
            },
            {
                "endpoints": [
                    {
                        "adminURL": "http://hostname1.domain:8776/v1/c8854bd3d15f4f479476d
                        "internalURL": "http://hostname1.domain:8776/v1/c8854bd3d15f4f4794
                        "publicURL": "http://hostname1.domain:8776/v1/c8854bd3d15f4f479476
                        "region": "RegionOne"
                    }
                ],
                "endpoints_links": [],
                "name": "volume",
                "type": "volume"
            },
            {
                "endpoints": [
                    {
                        "adminURL": "http://hostname1.domain:8773/services/Admin",
                        "internalURL": "http://hostname1.domain:8773/services/Cloud",
                        "publicURL": "http://hostname1.domain:8773/services/Cloud",
                        "region": "RegionOne"
                    }
                ],
                "endpoints_links": [],
```

```
                "name": "ec2",
                "type": "ec2"
            },
            {
                "endpoints": [
                    {
                        "adminURL": "http://hostname1.domain:8888/v1/AUTH_c8854bd3d15f4f47
                        "internalURL": "http://hostname1.domain:8888/v1/AUTH_c8854bd3d15f4
                        "publicURL": "http://hostname1.domain:8888/v1/AUTH_c8854bd3d15f4f4
                        "region": "RegionOne"
                    }
                ],
                "endpoints_links": [],
                "name": "swift",
                "type": "object-store"
            },
            {
                "endpoints": [
                    {
                        "adminURL": "http://hostname1.domain:35357/v2.0/",
                        "internalURL": "http://hostname1.domain:5000/v2.0/",
                        "publicURL": "http://hostname1.domain:5000/v2.0/",
                        "region": "RegionOne"
                    }
                ],
                "endpoints_links": [],
                "name": "keystone",
                "type": "identity"
            }
        ],
        "token": {
            "expires": "2012-10-11T09:49:27Z",
            "id": "1d9c15ed60414a92bc39d3b989d5bcae",
            "tenant": {
                "description": "Admin Tenant Description",
                "enabled": true,
                "id": "c8854bd3d15f4f479476d0cf58ef7db2",
                "name": "adminTenant"
            }
        },
        "user": {
            "id": "4b6da6bfa3634c819fc1abc022e88a22",
            "name": "adminUser",
            "roles": [
                {
                    "id": "ad6dc79aa39c4249ab63840f9591f99d",
                    "name": "admin"
                }
            ],
            "roles_links": [],
            "username": "adminUser"
        }
    }
}
</verbatim

-- Main.MatteoManzali - 2012-10-10
%META:TOPICMOVED{by="BaseUserMapping_333" date="1350915059" from="GestioneServiziGenerali.
</div><!-- /patternTopic-->

%META{"form"}%
%META{"attachments"}%</div><!-- /patternContent-->
---
%MAKETEXT{"This topic:"}% <nop>%WEB%%META{"parent" prefix="<span class='twikiSeparator'>&n
%MAKETEXT{"Topic revision:"}% %REVINFO{format="r$rev - $date - <nop>$wikiname"}%
</div><!-- /patternMainContents-->
</div><!-- /patternMain-->
```

```
</div><!-- /patternFloatWrap-->
<div class="clear"> </div>
</div><!-- /patternOuter--><div id="patternBottomBar"><div id="patternBottomBarContents"><
</div><!-- /patternPage-->
</div><!-- /patternPageShadow-->
</div><!-- /patternScreen-->
</body></html>
```