

OpenStack: How do I configure Bright OpenStack 7.2 with Active Directory authentication?

Bright OpenStack 7.2 with AD authentication

This Knowledge base article describes the steps needed to successfully configure Bright OpenStack 7.2 to use the Active Directory/LDAP backend.

It is now possible to configure AD/LDAP authentication from cmsh in a very simple way.

The LDAP backend will be used for the user/group identity and the MySQL backend will be used for role assignment.

Steps:

Set the admin token:

```
# cmsh -c "openstack; settings; credentials; set admintoken `openssl rand -hex 10`;commit"
```

Remove the SQL backend and add an LDAP backend and configure the LDAP backend to bind to Active Directory, change the <LDAP URL> to the URL of the Active Directory server, eg. ldap://ad.brightcomputing.com. Then create a service user in Active Directory and set the username and the password:

```
# cmsh -c "openstack; settings; authentication; authbackends; remove sql; add ldap ad; set url <LDAP URL>; set username <USERNAME>; set password <PASSWORD>; commit"
```

Configure the attribute, objectclass, and the base dn for searching the user/group, eg. CN=Users,DC=bright,DC=com.

```
# cmsh -c "openstack; settings; authentication; authbackends; use ad; usersettings; set idattribute sAMAccountName; set nameattribute sAMAccountName; set objectclass person; set treedn <SEARCH TREE>; commit"
```

```
# cmsh -c "openstack; settings; authentication; authbackends; use ad; groupsettings; set idattribute sAMAccountName; set memberattribute member; set objectclass group; set treedn <SEARCH TREE>; commit"
```

OpenStack: How do I configure Bright OpenStack 7.2 with Active Directory authentication?

Configure each service user and password used by the OpenStack component.

In this case each service user created in the previous step is used for each OpenStack user.

It is a best practice to use a user/password for each service, as is done by default with the SQL backend.

Retrieve the username/password pair, and create each user with the associated password in Active Directory.

```
# cmsh -c 'openstack settings; credentials; get cinderusername; get cinderpassword'

# cmsh -c 'openstack settings; credentials; get keystoneusername; get keystonepassword'

# cmsh -c 'openstack settings; credentials; get cmdaemonopenstackusername; get cmdaemonopenstackpassword'

# cmsh -c 'openstack settings; credentials; get glanceusername; get glancepassword'

# cmsh -c 'openstack settings; credentials; get heatusername; get heatpassword'

# cmsh -c 'openstack settings; credentials; get neutronusername; get neutronpassword'

# cmsh -c 'openstack settings; credentials; get novausername; get novapassword'

# echo admin && cmsh -c 'openstack settings; credentials; get mainadminpassword'
```

If just one username/password pair will be used, then change <USERNAME> and <PASSWORD> accordingly as in the following:

```
# (for i in {cinder,glance,heat,keystone,neutron,nova,radosgw,cmdaemonopenstack}; do echo "openstack use default; settings; credentials; set ${i}username <USERNAME>; home"; done; echo openstack commit ) | cmsh
```

OpenStack: How do I configure Bright OpenStack 7.2 with Active Directory authentication?

```
# (for i in {cinder,glance,heat,keystone,neutron,nova,radosgw,cmdaemonopenstack}; do echo "openstack use default; settings; credentials; set ${i}password <PASSWORD>; home"; done; echo openstack commit ) | cmsh
```

Assign the admin role to the user created before for the bright and service project, and replace <USERNAME> accordingly.

Note that if a different username/password is used for each service, then the step must be carried out for each user.

```
# export TOKEN=$(cmsh -c 'openstack; settings; credentials; get admintoken')
# openstack --os-token $TOKEN --os-url http://master:5000/v3 role add --user <USERNAME> --project bright admin
# openstack --os-token $TOKEN --os-url http://master:5000/v3 role add --user <USERNAME> --project service admin
```

Using the admin user created before in AD, log in to OpenStack and assign the project membership.

Unique solution ID: #1304

Author: Matteo Piccinini

Last update: 2016-04-13 17:41