The LDAP plugin for Fuel documentation

Release 2.0-2.0.0-1

Mirantis Inc.

May 12, 2016

CHAPTER

ONE

PLUGIN GUIDE

1.1 LDAP plugin for Fuel

This plugin extends Mirantis OpenStack functionality by adding LDAP support. It allows to use an existing LDAP server as authentication backend for Keystone. Enabling this plugin means that all users except system users will be authenticated against the configured LDAP server.

Please note that Fuel will not validate the settings, e.g. by attempting to connect to the LDAP server.

1.1.1 Requirements

Requirement	Version/Comment
Fuel	8.0
Pre-configured LDAP server	

LDAP server should be pre-deployed and be accessible via Public network from Controller nodes.

1.2 Release notes / Changelog

2.0.0

- Support of multi-domains
- Compatibility with MOS 8.0

1.0.0

- This is the first release of the plugin

1.3 LDAP plugin limitations

1. LDAP plugin has the following limitations:
    - Installation of LDAP plugin before deployment only;
    - Fuel will not validate the settings, e.g., by attempting to connect to the LDAP server;
    - In multidomain configuration the attributes of the first domain are filled in the web form, whereas the attributes of other domains are filled in one field;
    - The settings of domains determined in “List of additional Domains” field will not be validated;

1.4 Installation Guide

1.4.1 Installing LDAP plugin

To install LDAP plugin, follow these steps:

1. Download the plugin from the Fuel Plugins Catalog.
2. Copy the plugin on an already installed Fuel Master node (SSH can be used for that). If you do not have the Fuel Master node yet, see Quick Start Guide:

```
# scp ldap-2.0-2.0.0-1.noarch.rpm root@<Fuel_Master_IP>:/tmp
```

3. Log into the Fuel Master node. Install the plugin:

```
# cd /tmp
# fuel plugins --install ldap-2.0-2.0.0-1.noarch.rpm
```

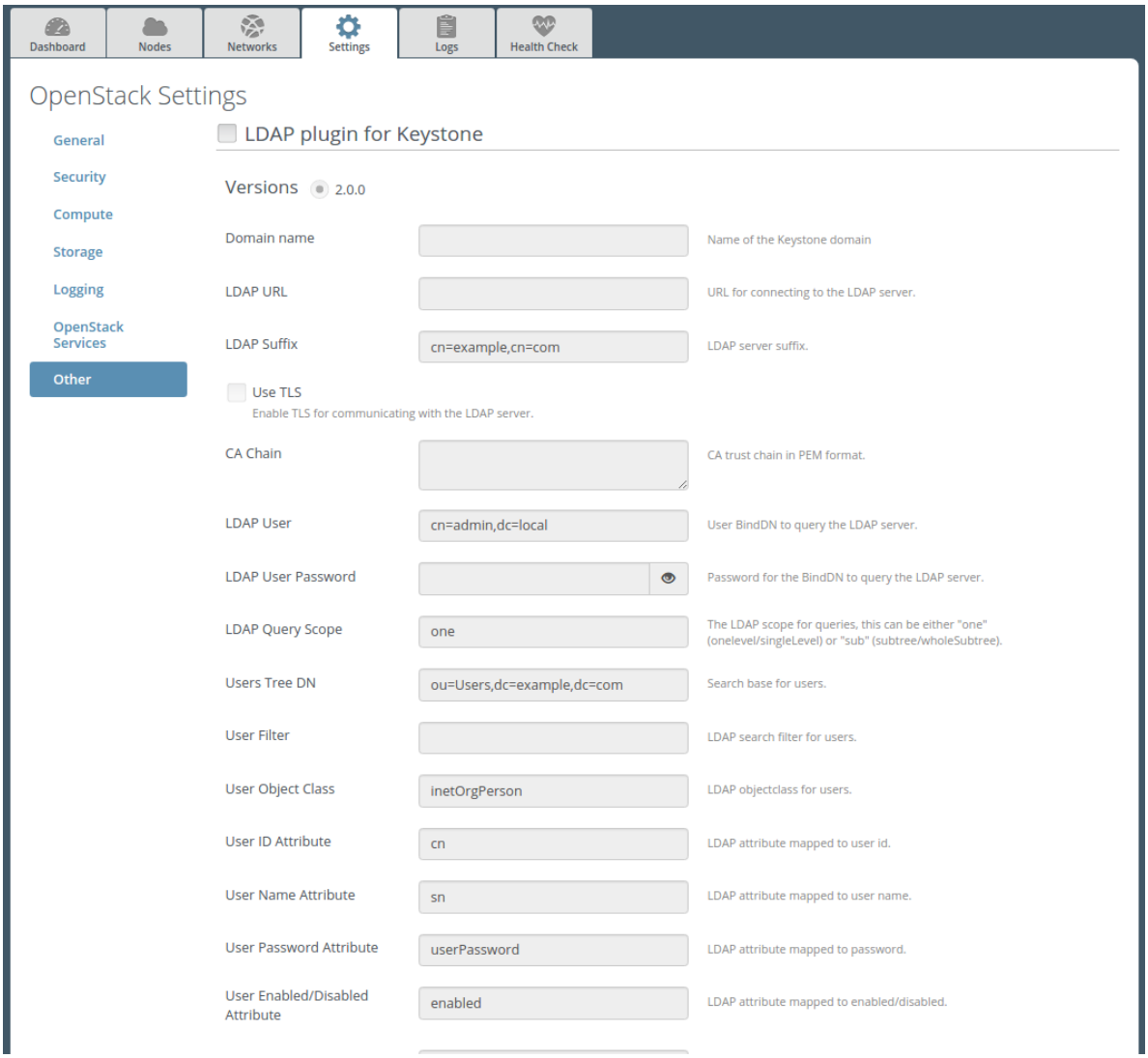
4. Check if the plugin was installed successfully

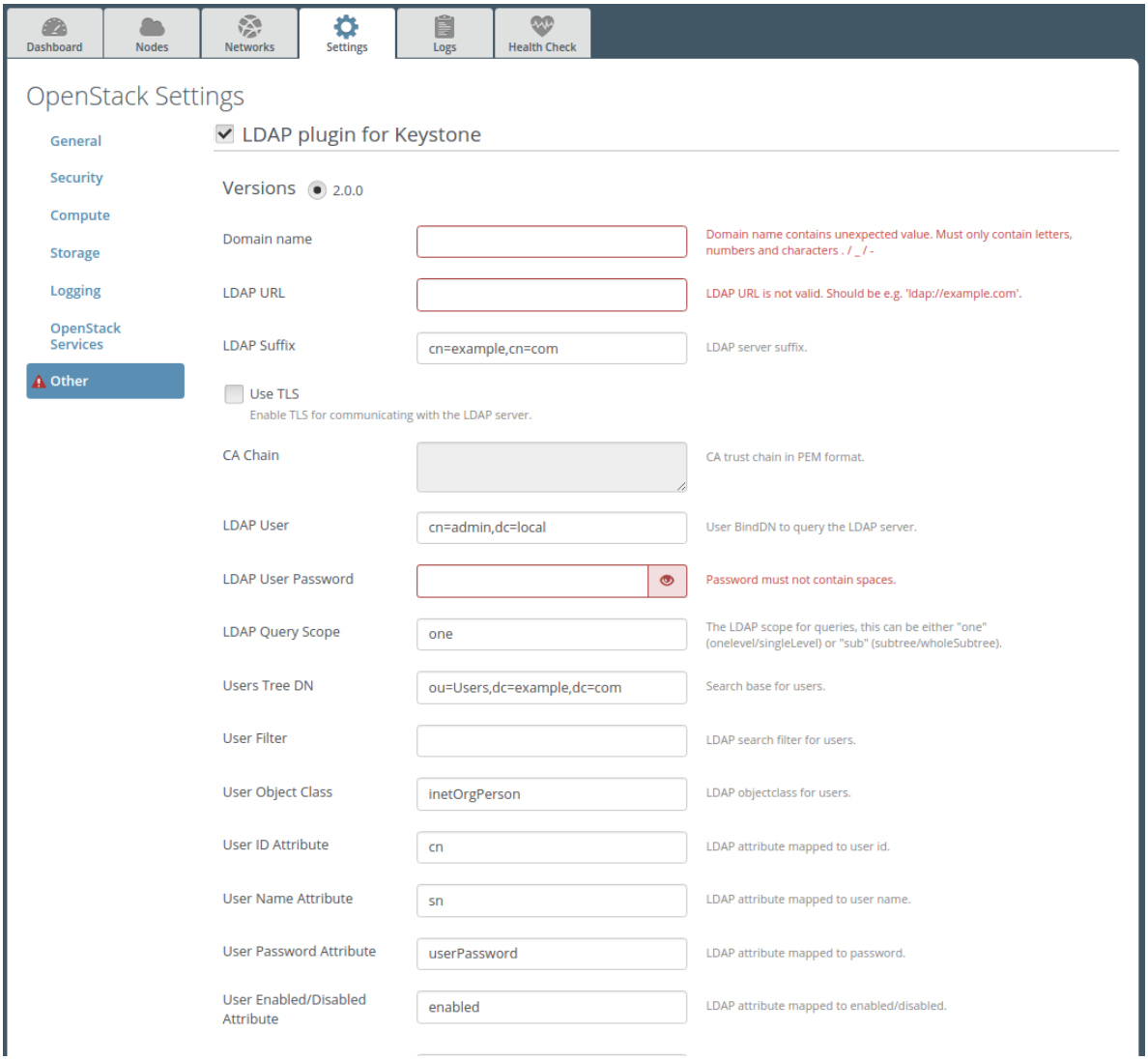
```
# fuel plugins
id | name         | version  | package_version
---|--------------|----------|----------------
1  | ldap         | 2.0.0    | 3.0.0
```

5. MU-1 (Maintenance Update) should be installed to provide proper work of keystone providers ith domains during deployment process.

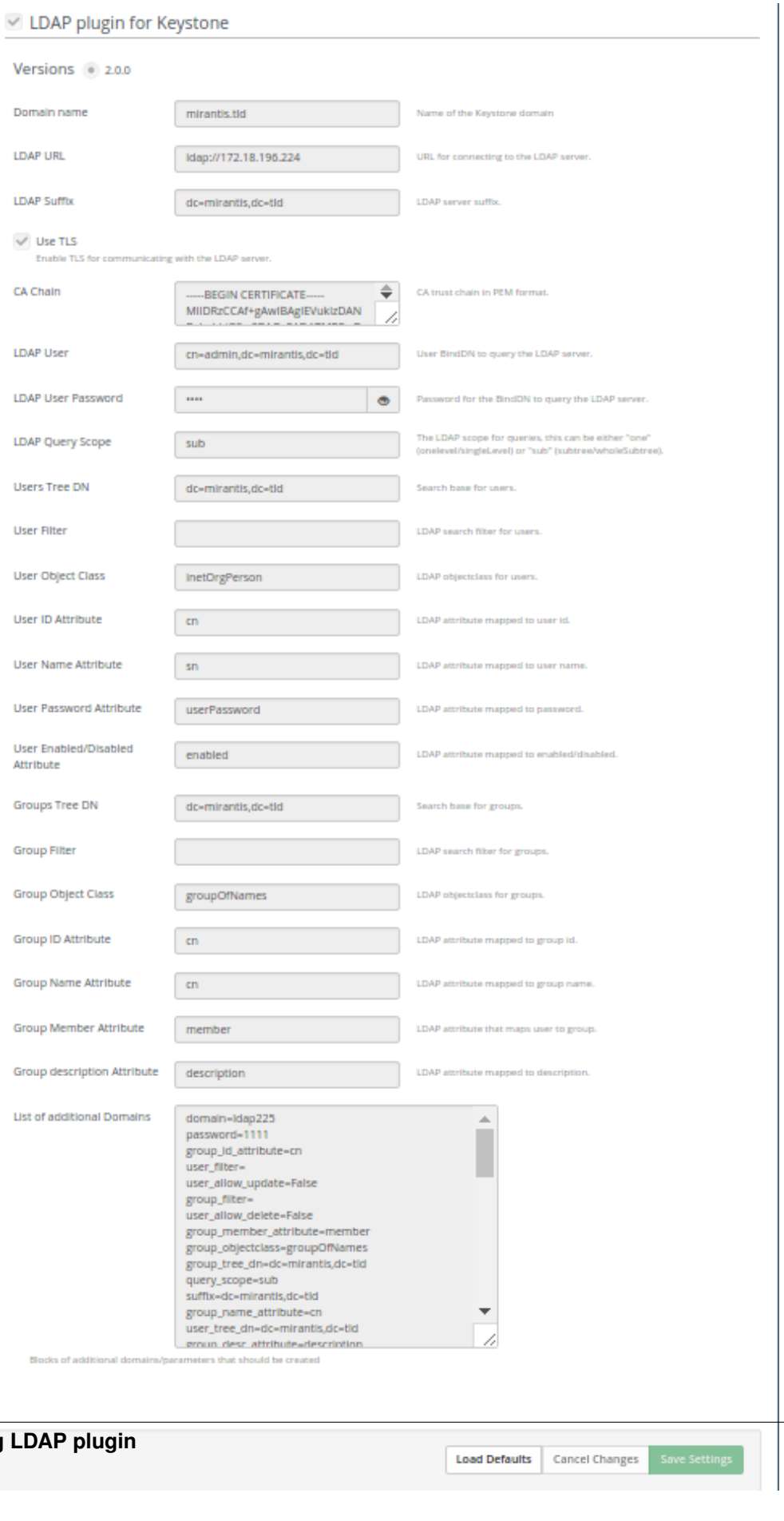
1.5 Configuring LDAP plugin

1. Create a new OpenStack environment to use an existing LDAP server as authentication backend for Keystone. For more information about environment creation, see Mirantis OpenStack User Guide.
2. Open *Settings* tab of the Fuel Web UI, scroll the page down and select the *LDAP plugin for Keystone* checkbox:

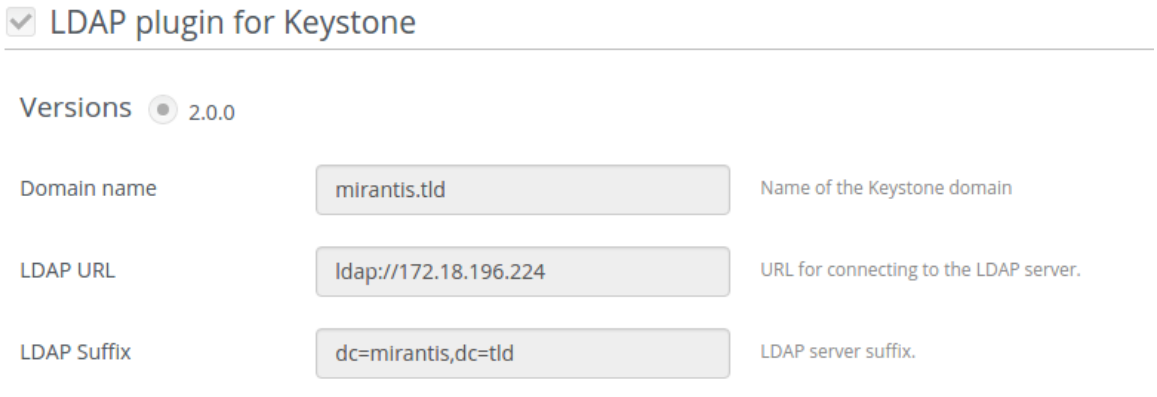

Dashboard
Nodes
Networks
Settings
Logs
Health Check
OpenStack Settings
General
Security
Compute
Storage
Logging
OpenStack Services
Other
LDAP plugin for Keystone
Versions
2.0.0
Domain name
Name of the Keystone domain
LDAP URL
URL for connecting to the LDAP server.
LDAP Suffix
cn=example,cn=com
LDAP server suffix.
Use TLS
Enable TLS for communicating with the LDAP server.
CA Chain
CA trust chain in PEM format.
LDAP User
cn=admin,dc=local
User BindDN to query the LDAP server.
LDAP User Password
Password for the BindDN to query the LDAP server.
LDAP Query Scope
one
The LDAP scope for queries, this can be either "one" (onelevel/singleLevel) or "sub" (subtree/wholeSubtree).
Users Tree DN
ou=Users,dc=example,dc=com
Search base for users.
User Filter
LDAP search filter for users.
User Object Class
inetOrgPerson
LDAP objectclass for users.
User ID Attribute
cn
LDAP attribute mapped to user id.
User Name Attribute
sn
LDAP attribute mapped to user name.
User Password Attribute
userPassword
LDAP attribute mapped to password.
User Enabled/Disabled Attribute
enabled
LDAP attribute mapped to enabled/disabled.



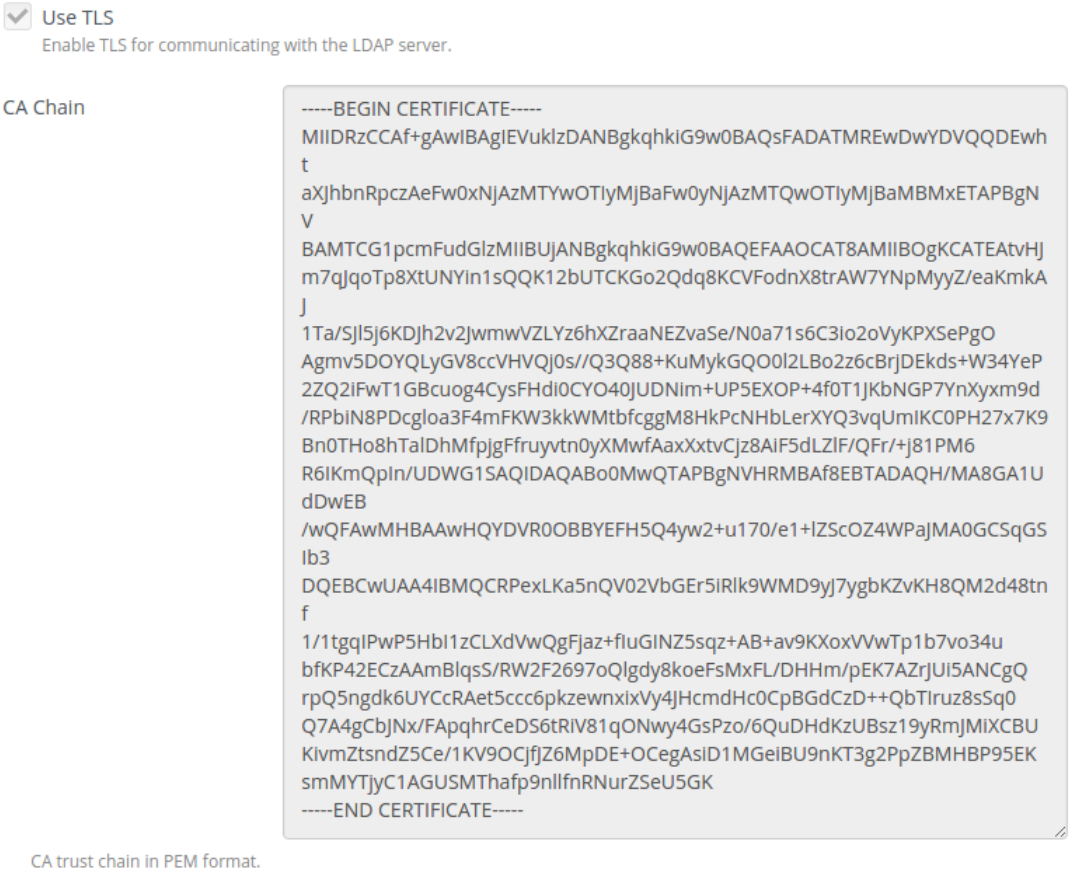
3. Enter plugin settings into the text fields:


LDAP plugin for Keystone
Versions
2.0.0
Domain name
mirantis.tld
Name of the Keystone domain
LDAP URL
ldap://172.18.196.224
URL for connecting to the LDAP server.
LDAP Suffix
dc=mirantis,dc=tld
LDAP server suffix.
Use TLS
Enable TLS for communicating with the LDAP server.
CA Chain
-----BEGIN CERTIFICATE-----
MIIDRzCCAf+gAwIBAgIEVuklzDAN
CA trust chain in PEM format.
LDAP User
cn=admin,dc=mirantis,dc=tld
User BindDN to query the LDAP server.
LDAP User Password
••••
Password for the BindDN to query the LDAP server.
LDAP Query Scope
sub
The LDAP scope for queries, this can be either "one" (onelevel/singleLevel) or "sub" (subtree/wholeSubtree).
Users Tree DN
dc=mirantis,dc=tld
Search base for users.
User Filter
LDAP search filter for users.
User Object Class
inetOrgPerson
LDAP objectclass for users.
User ID Attribute
cn
LDAP attribute mapped to user id.
User Name Attribute
sn
LDAP attribute mapped to user name.
User Password Attribute
userPassword
LDAP attribute mapped to password.
User Enabled/Disabled Attribute
enabled
LDAP attribute mapped to enabled/disabled.
Groups Tree DN
dc=mirantis,dc=tld
Search base for groups.
Group Filter
LDAP search filter for groups.
Group Object Class
groupOfNames
LDAP objectclass for groups.
Group ID Attribute
cn
LDAP attribute mapped to group id.
Group Name Attribute
cn
LDAP attribute mapped to group name.
Group Member Attribute
member
LDAP attribute that maps user to group.
Group description Attribute
description
LDAP attribute mapped to description.
List of additional Domains
domain=ldap225
password=1111
group_id_attribute=cn
user_filter=
user_allow_update=False
group_filter=
user_allow_delete=False
group_member_attribute=member
group_objectclass=groupOfNames
group_tree_dn=dc=mirantis,dc=tld
query_scope=sub
suffix=dc=mirantis,dc=tld
group_name_attribute=cn
user_tree_dn=dc=mirantis,dc=tld
Blocks of additional domains/parameters that should be created
Load Defaults
Cancel Changes
Save Settings

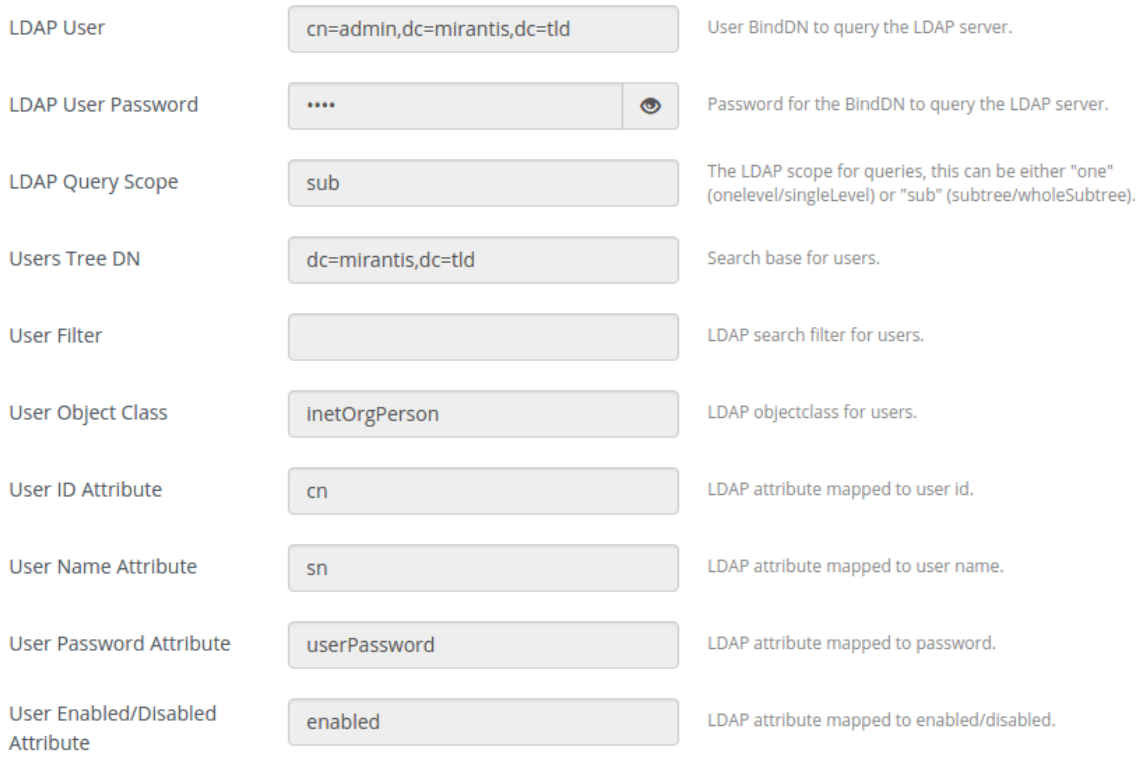
Specify domain name, LDAP URL, LDAP suffix:



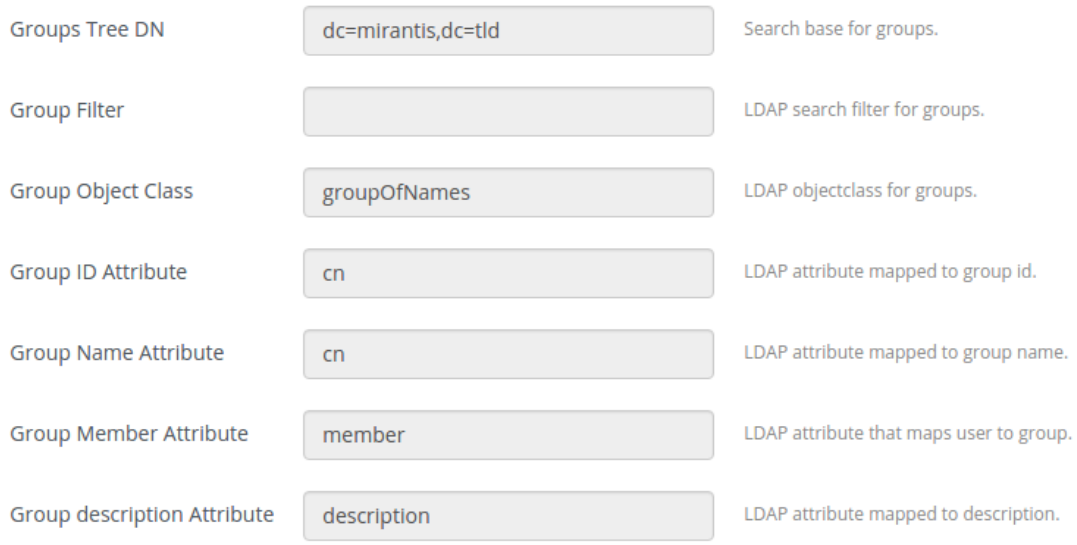
Enable TLS use and put certificate if it is needed:



Specify LDAP user, password and other settings:



To use LDAP groups provide settings for it:



Fields description:

Field	Comment
Domain name	Name of the Keystone domain.
LDAP URL	URL for connecting to the LDAP server.
LDAP Suffix	LDAP server suffix.
Use TLS	Enable TLS for communicating with the LDAP server.
CA Chain	CA trust chain in PEM format.
LDAP User	User BindDN to query the LDAP server.
LDAP User Password	Password for the BindDN to query the LDAP server.
LDAP Query Scope	The LDAP scope for queries, this can be either "one" (onelevel/singleLevel) or "sub" (subtree/wholeSubtree).
Users Tree DN	Search base for users.
User Filter	LDAP search filter for users.
User Object Class	LDAP objectclass for users.
User ID Attribute	LDAP attribute mapped to user id.
User Name Attribute	LDAP attribute mapped to user name.
User Password Attribute	LDAP attribute mapped to password.
User Enabled/Disabled Attribute	LDAP attribute mapped to enabled/disabled.
Groups Tree DN	Search base for groups.
Group Filter	LDAP search filter for groups.
Group Object Class	LDAP objectclass for groups.
Group ID Attribute	LDAP attribute mapped to group id.
Group Name Attribute	LDAP attribute mapped to group name.
Group Member Attribute	LDAP attribute that maps user to group.
Group description Attribute	LDAP attribute mapped to description.
List of additional Domains	Blocks of additional domains/parameters that should be created.

4. To deploy an environment with support of multiple domains 'List of additional Domains' text area should be used. All needed parameters that describes a domain should be copied there, all parameters form a block of parameters.

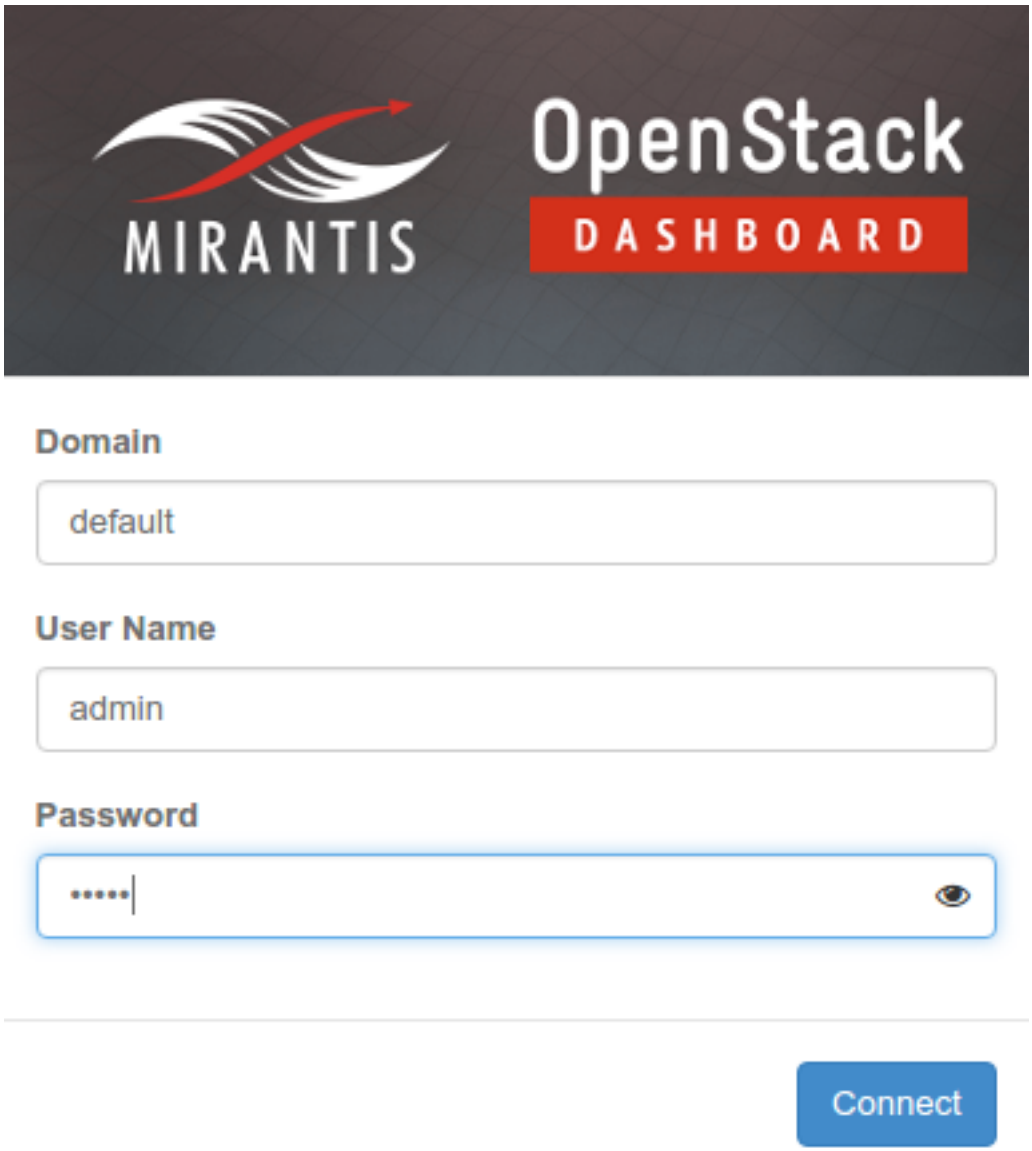
List of additional Domains

```
domain=ldap225
password=1111
group_id_attribute=cn
user_filter=
user_allow_update=False
group_filter=
user_allow_delete=False
group_member_attribute=member
group_objectclass=groupOfNames
group_tree_dn=dc=mirantis,dc=tld
query_scope=sub
suffix=dc=mirantis,dc=tld
group_name_attribute=cn
user_tree_dn=dc=mirantis,dc=tld
group_desc_attribute=description
url=ldap://172.18.196.224
user_allow_create=False
user_id_attribute=cn
user_pass_attribute=userPassword
tls_cacertdir=/etc/ssl/certs
group_allow_delete=False
group_allow_create=False
user=cn=admin,dc=mirantis,dc=tld
user_enabled_attribute=enabled
use_tls=True
user_objectclass=inetOrgPerson
group_allow_update=False
user_name_attribute=sn
ca_chain=-----BEGIN CERTIFICATE-----
MIIDRzCCAf+gAwIBAgIEVuklzDANBgkqhkiG9w0BAQsFADATMREwDwYDVQQDE
wht
aXJhbnRpczAeFw0xNjAzMTYwOTIyMjBaFw0yNjAzMTQwOTIyMjBaMBMxETAPB
gNV
BAMTCG1pcmFudGlzMIIBUjANBgkqhkiG9w0BAQEFAAOCAT8AMIIBOgKCATEAt
vHJ
m7qJqoTp8XtUNYin1sQQK12bUTCKGo2Qdq8KCVFodnX8trAW7YNpMyyZ/eaK
mkAJ
1Ta/SJI5j6KDJh2v2JwmwVZLYz6hXZraaNEZvaSe/N0a71s6C3io2oVyKPXSePgO
Agmv5DOYQLyGV8ccVHVQj0s//Q3Q88+KuMykGQO0I2LBo2z6cBrjDEkds+W34Y
eP
2ZQ2iFwT1GBcuog4CysFHdi0CYO40JUDNim+UP5EXOP+4f0T1JKbNGP7YnXyxm
9d
/RPbiN8PDcgloa3F4mFKW3kkWMtbfcggM8HkPcNHbLerXYQ3vqUmIKC0PH27x
7K9
Bn0THo8hTaIDhMfpjgFfruyvtn0yXMwfAaxXxtvCjz8AiF5dLZIF/QFr/+j81PM6
R6IKmQpIn/UDWG1SAQIDAQABo0MwQTAPBgNVHRMBAf8EBTADAQH/MA8GA
1UdDwEB
/wQFAwMHBAAwHQYDVR0OBBYEFH5Q4yw2+u170/e1+IZScOZ4WPaJMA0GCS
qGSIb3
DQEBCwUAA4IBMQCRPexLKa5nQV02VbGEr5iRIk9WMD9yJ7ygbKZvKH8QM2d4
8tnf
1/1tgqIPwP5HbI1zCLXdVwQgFjaz+fluGINZ5sqz+AB+av9KXoxVVwTp1b7vo34u
bfKP42ECzAAmBIqsS/RW2F2697oQlgdy8koeFsMxFL/DHHm/pEK7AZrJUi5ANCg
Q
rpQ5ngdk6UYCcRAet5ccc6pkzewnxixVy4JHcmdHc0CpBGdCzD++QbTIruz8sSq0
Q7A4gCbJNx/FApqhrCeDS6tRiV81qONwy4GsPzo/6QuDHdKzUBsz19yRmJMiXC
BU
KivmZtsndZ5Ce/1KV9OCjfJZ6MpDE+OCegAsiD1MGeiBU9nKT3g2PpZBMHBP95
EK
smMYTjyC1AGUSMThafp9nIlfnRNurZSeU5GK
-----END CERTIFICATE-----
```

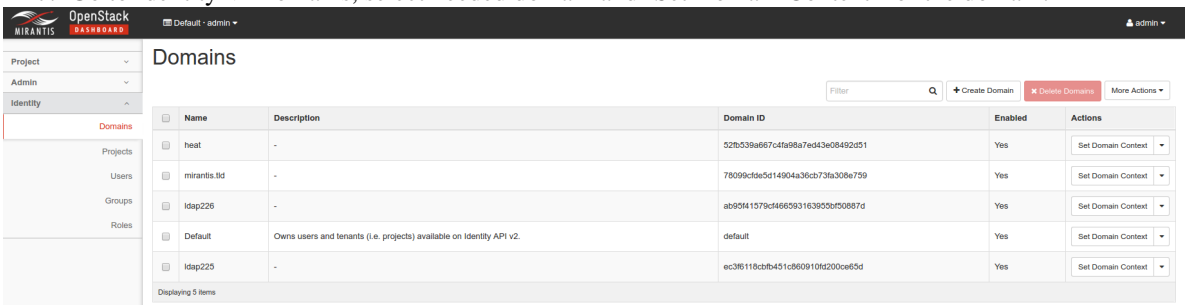
Blocks of additional domains/parameters that should be created

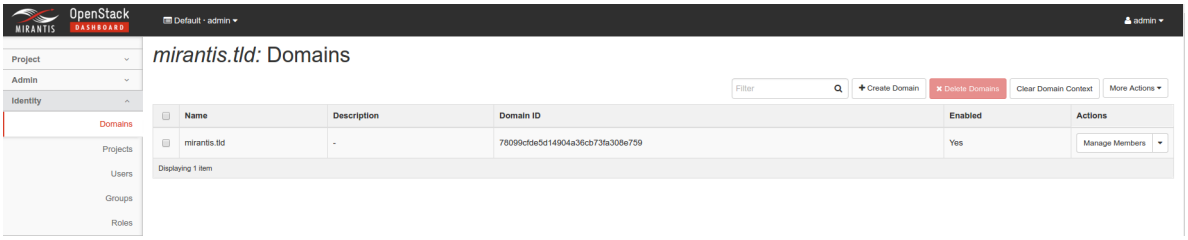
To add multiple domains such block of parameters should be added to 'List of additional Domains' text area and these blocks should be separated by empty line.

5. Continue with environment configuration and deploy it; for instructions, see Mirantis OpenStack User Guide.
6. After successfull environment deployment log into dashboard in default domain:

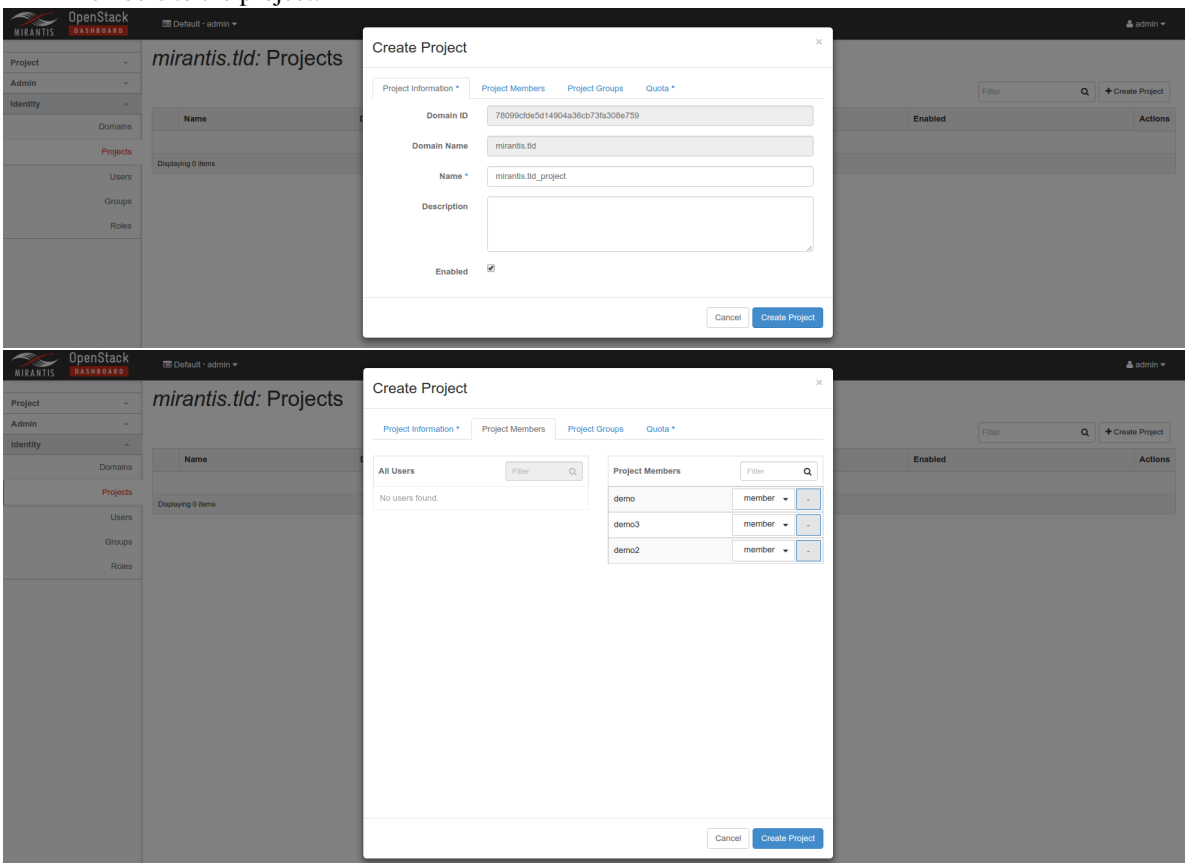


7. Go to Identity -> Domains, select needed domain and 'Set Domain Context' for the domain:



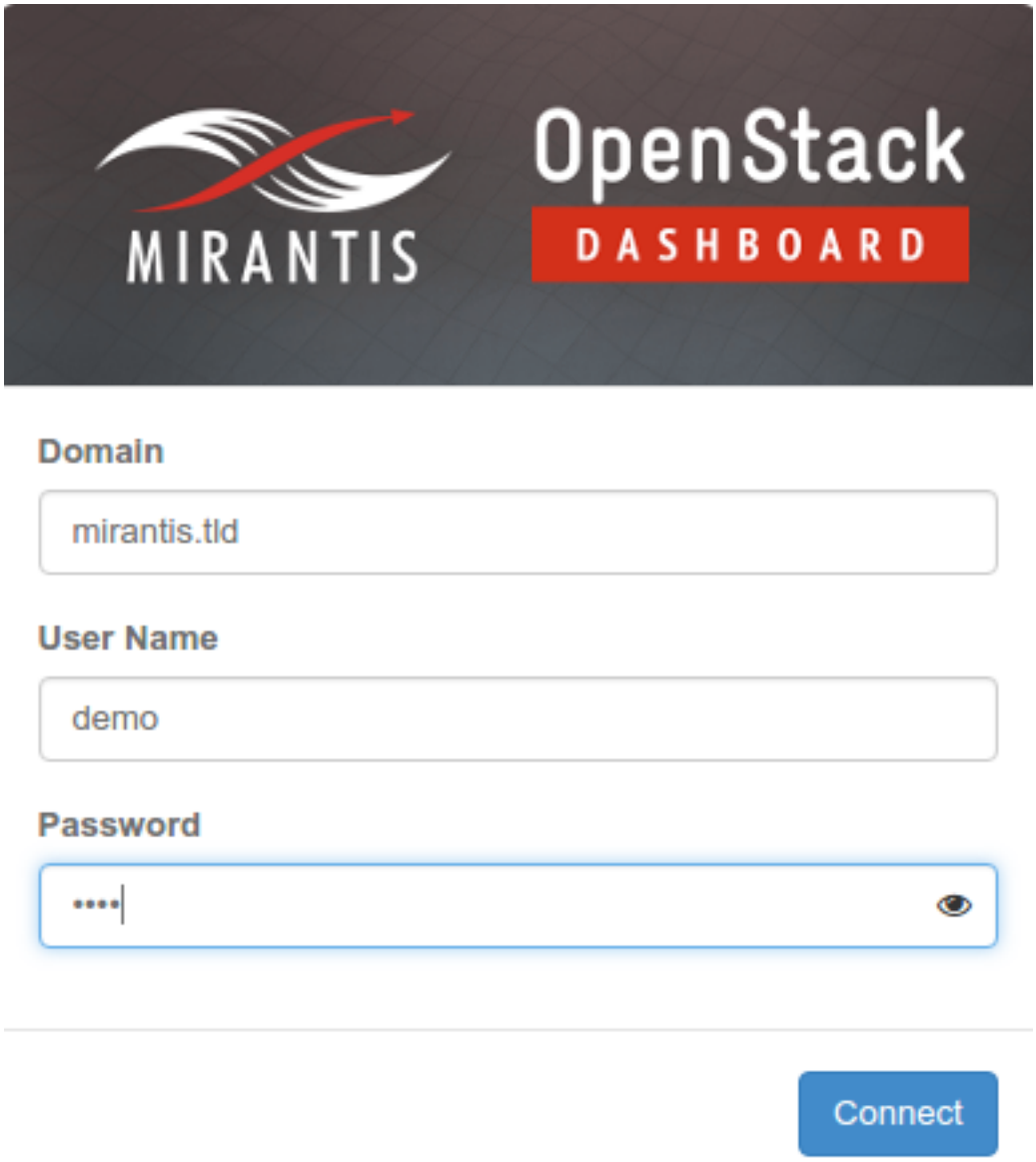


8. Go to Identity -> Projects and select 'Create Project' to create a new project for the domain and add user members to the project:



1.6 User Guide

1. After successful deployment, all users from the LDAP directory matching the configured filter criteria can authenticate against Keystone. To validate the configuration, log into the Horizon dashboard using LDAP credentials:



1.7 LDAP plugin validation

1. To validate that LDAP plugin is successfully applied after deployment:
   - Log into Horizon using domain/user credentials from LDAP server;
   - Create an instance;

   Expecting results:
   - All LDAP users can authenticate via Keystone;
   - An instance is successfully created;

1.8 Troubleshooting

1.8.1 Checking presence of LDAP domain/users

To get a list of domains in keystone run the following command on Controller node:

```
OS_IDENTITY_API_VERSION=3 OS_AUTH_URL='<http://192.168.0.2:5000/v3/>' openstack domain list
```

To get a list of users in a domain run the following command on Controller node:

```
  OS_IDENTITY_API_VERSION=3 OS_AUTH_URL='<http://192.168.0.2:5000/v3/>' openstack user list --quiet -

where 'http://192.168.0.2:5000/v3/' is internal keystone url.
```

1.8.2 Checking LDAP server availability

To check LDAP server availability run the following command on Controller node:

```
ldapsearch -H ldap://<url/ip_address> -x -b dc=<ldap>,dc=<suffix>
```

1.8.3 LDAP plugin log files

As LDAP plugin only updates keystone configuration files to check keystone service, these files keep logs:

/var/log/apache2/keystone_wsgi_admin_access.log

/var/log/apache2/keystone_wsgi_admin_error.log

/var/log/apache2/keystone_wsgi_main_access.log

/var/log/apache2/keystone_wsgi_main_error.log

1.9 Appendix

1.9.1 Links

- Mirantis OpenStack Documentation Center
- Fuel Plugins Catalog