

Murano User Guide

Murano User Guide

v0.1

Publication date 2013-09-04
Copyright © 2013 Mirantis, Inc.

Abstract

This document is intended for individuals who wish to use Murano Product.

Table of Contents

1. How can I use Murano Service?	1
Document change history	1
2. Murano dashboard plugin	2
Creating environment	2
Creating service prototype	2
Active Directory	4
Internet Information Service	6
Internet Information Web Farm Service	7
ASP.NET Service	8
ASP.NET Farm Service	9
MS SQL Service	10
SQL Server Failover Cluster	11
Deploying environment	14
Working with deployed environment	16
Redeploying Murano Environment	18
Deleting	18
Browsing Murano Deployment	19
Renaming Murano Environment	21
Troubleshooting	22

Chapter 1. How can I use Murano Service?

Murano is intended to get opportunity for non-experienced users to deploy reliable Windows-based environments extremely simple. This document describes process of creation virtual Windows Environment with different services.

Document change history

The following table describes the most recent changes:

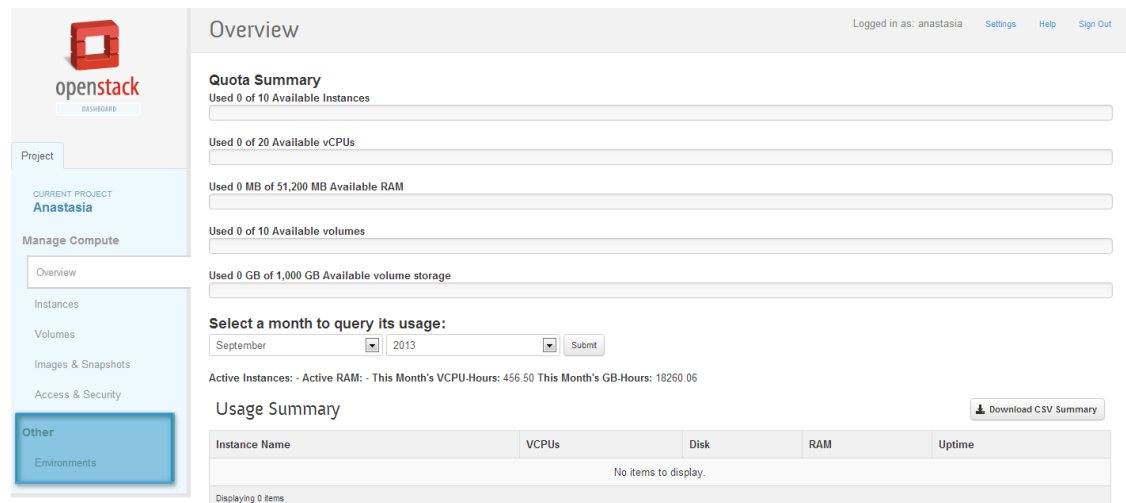
Revision Date	Summary of Changes
September. 4, 2013	• Initial document creation.

Chapter 2. Murano dashboard plugin

To use Murano you should be familiar with Openstack. Murano Dashboard is just a plugin to Openstack dashboard - Horizon. Please visit [horizon user guide](http://docs.openstack.org/user-guide/content/) [http://docs.openstack.org/user-guide/content/] first to see how dashboard is organized and how to login to it.

Creating environment

Once you installed all Murano components and login to horizon dashboard successfully you will see Environments panel:



Overview

Logged in as: anastasia Settings Help Sign Out

Quota Summary

Used 0 of 10 Available Instances

Used 0 of 20 Available vCPUs

Used 0 MB of 51,200 MB Available RAM

Used 0 of 10 Available volumes

Used 0 GB of 1,000 GB Available volume storage

Select a month to query its usage:

September 2013 Submit

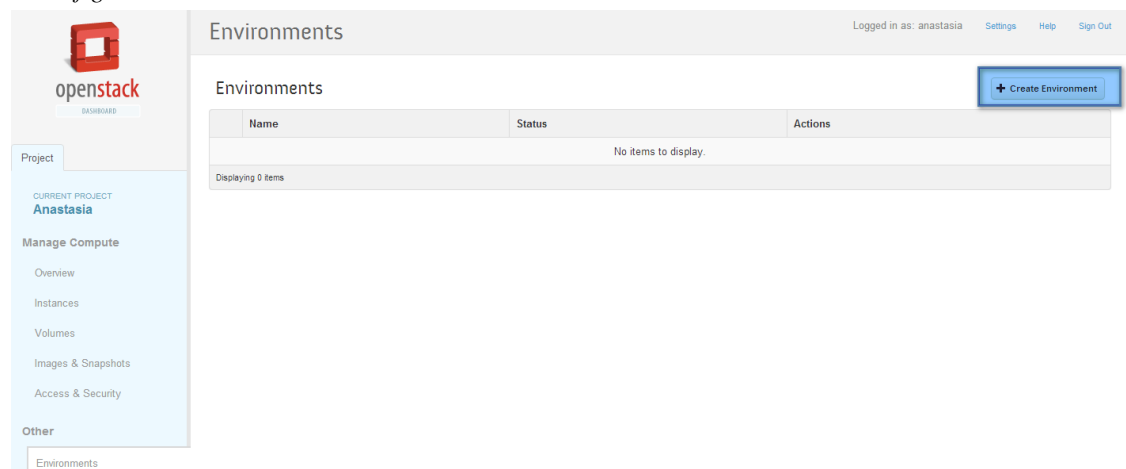
Active Instances: - Active RAM: - This Month's VCPU-Hours: 456.50 This Month's GB-Hours: 18260.06

Usage Summary Download CSV Summary

Instance Name	VCPUs	Disk	RAM	Uptime
No items to display.				

Displaying 0 items

First thing you need to do is to create an environment - virtual Windows Data Center which will contain different Windows services. Navigate to the "Environments" page and click the "Create Environment". After setting name to your virtual environment it will be created. Just created environment has status *Ready to configure*.



Environments

Logged in as: anastasia Settings Help Sign Out

Environments + Create Environment

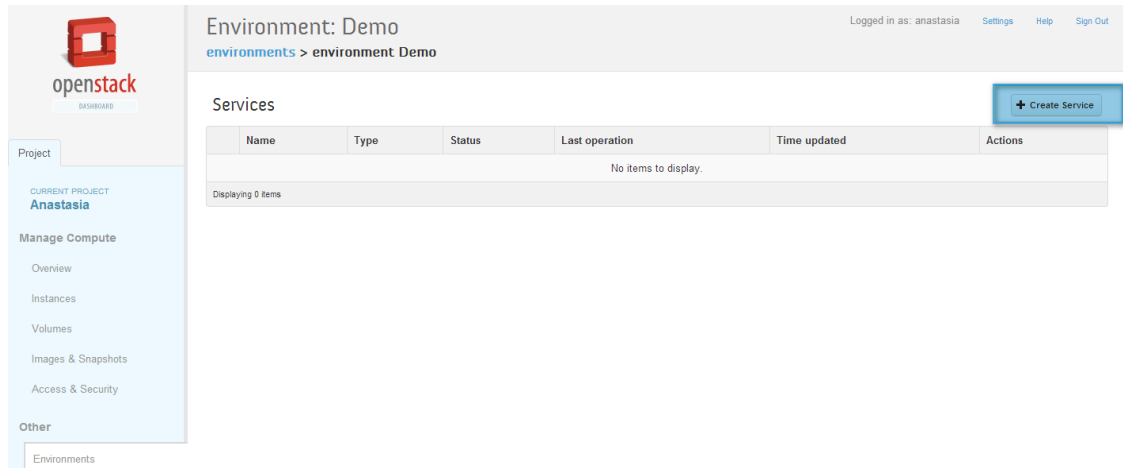
Name	Status	Actions
No items to display.		

Displaying 0 items

Creating service prototype

All services should be created in the context of Environment - virtual Windows Data Center. After Environment is created services prototypes and then deploy the Environment. When deploy process is

done instances with your services will be spawned in Openstack. To create a service prototype navigate to environment services by clicking on the environment name (or on the "Services" button) and click the "Create Service" button.



You have an opportunity to create one of the following services:

Create Service ✕

Service Type

- ASP.NET Application
- ASP.NET Application**
- ASP.NET Application Web Farm
- Active Directory
- Internet Information Services
- Internet Information Services Web Farm
- MS SQL Server
- MS SQL Server Cluster

Description:

The ASP.NET Application Service installs custom application onto one IIS Web Server

If you want your services to work with AD you should create The Active Directory Service first

Next

Once you choose service that you want to create click "Next" and fill the form. Forms for each service are specific. To see more information about filling the form for a specific service follow one of the link below:

- **Active Directory:** Active Directory is a directory service implemented by Microsoft for Windows domain networks. In one installation, in addition to primary Domain Controller, you can add optional count of secondary Domain Controllers. Any other services you are intending to create can be joined to that domain.
- **Internet Information Service:** IIS is a web server and a set of feature extension modules.
- **Internet Information Web Farm Service:** Murano installs the Web Farm Framework on the controller server, configures the primary server and prepares the secondary servers. In addition load balancer is installed to monitor service statuses.
- **ASP.NET Service:** is a server-side Web application framework designed for Web development to produce dynamic Web pages. Service is able to install custom application onto one IIS Web Server. Murano installs all needed components and make proper configuration.
- **ASP.NET Farm Service:** ASP.NET Farm Service installs a custom application on a load-balanced array of IIS servers
- **MS SQL Service:** Microsoft SQL Service is a relational database management system.
- **SQL Server Failover Cluster:** Murano installs all needed components and configures your SQL Server Cluster the way you want.

On the last step of creating service prototype you have opportunity to set the hardware flavor of the instance which will be created - and the image with the operating system, which will be installed on the instance. Also you may select availability zone, if there are more then one in your environment.

Create Service ×

Instance flavor

m1.medium

Instance image

Windows Server 2012 Standard

Availability zone

nova

Internet Information Services Service

Instance Configuration: Specify some instance parameters on which service would be created.

Instance flavor: Select registered in Openstack flavor. Consider that service performance depends on this parameter.

Instance image: Select valid image for a service. Image should already be prepared and registered in glance.

Availability zone: Select availability zone where service would be installed.

Back

Create

Active Directory

After you picked the Active Directory service in service creation wizard you'll see the following form:

Create Service ✕

Domain Name

Active Directory Service

Domain Name: Enter a desired name for a new domain. This name should fit to DNS Domain Name requirements: it should contain only A-Z, a-z, 0-9, (.) and (-) and should not end with a dash. DNS server will be automatically set up on each of the Domain Controller instances. Note: Only first 15 characters or characters before first period is used as NetBIOS name.

Instance Count: You can create several Active Directory instances by setting instance number larger than one. One primary Domain Controller and a few secondary DCs will be created.

Passwords: Windows requires strong password for service administration. Your password should have at least one letter in each register, a number and a special character. Password length should be a minimum of 7 characters. Once you forget your password you won't be able to operate the service until recovery password would be entered. So it's better for Recovery and Administrator password to be different.

Hostname template: For your convenience all instance hostnames can be named in the same way. Enter a name and use # character for incrementation. For example, host# turns into host1, host2, etc. Please follow Windows hostname restrictions.

Instance Count

Account Name

Administrator password

Confirm password

Recovery password

Confirm password

Hostname template

Back
Next

Enter or select values for this fields:

Domain name	Enter a desired name for a new domain. This name should fit to DNS Domain Name requirements: it should contain only A-Z, a-z, 0-9, (.) and (-) and should not end with a dash. DNS server will be automatically set up on each of the Domain Controller instances. Period characters are allowed only when they are used to delimit the components of domain style names. Single-level domain is not appropriate. Note: Only first 15 characters or characters before first period is used as NetBIOS name.
Instance Count	You can create several Active Directory instances by setting instance number larger than one. One primary Domain Controller and a few secondary DCs will be created.
Account Name	You account will have Active Directory administrator rights. So initial value is "Administrator" but you can change it to any name you like.
Administrator password	Windows requires strong password for service administration. Your password should have at least one letter in each register, a number and a special character. Password length should be a minimum of 7 characters.

	Once you forget your password you won't be able to operate the service until recovery password would be entered. So it's better for Recovery and Administrator password to be different.
Confirm password	Password confirmation is required.
Recovery password	Restrictions are the same as for the Administrator password. Please provide password that is different from Administrator. It's not required for form validation though.
Confirm password	Password confirmation is required.

Internet Information Service

Create Service ×

Service Name

Administrator password

Confirm password

Domain

Hostname template

Internet Information Services Service

Standalone IIS Server

Service Name: Enter a desired name for a service. Just A-Z, a-z, 0-9, dash and underline are allowed.

Passwords: Windows requires strong password for service administration. Your password should have at least one letter in each register, a number and a special character. Password length should be a minimum of 7 characters.

Domain: Service can be joined to the Active Directory domain. If you want to create an AD domain create the AD Service first.

Hostname template: For your convenience all instance hostnames can be named in the same way. Enter a name and use # character for incrementation. For example, host# turns into host1, host2, etc. Please follow Windows hostname restrictions.

Back

Next

Internet Information Web Farm Service

Create Service



Service Name

Administrator password

Confirm password

Domain

Instance Count

Load Balancer port

Hostname template

Internet Information Services Web Farm Service

A load-balanced array of IIS servers

Service Name: Enter a desired name for a service. Just A-Z, a-z, 0-9, dash and underline are allowed.

Passwords: Windows requires strong password for service administration. Your password should have at least one letter in each register, a number and a special character. Password length should be a minimum of 7 characters.

Domain: Service can be joined to the Active Directory domain. If you want to create an AD domain create the AD Service first.

Instance Count: Several instances with IIS Service can be created at one time.

Load Balancer port: Specify port number where Load Balancer will be running

Hostname template: For your convenience all instance hostnames can be named in the same way. Enter a name and use # character for incrementation. For example, host# turns into host1, host2, etc. Please follow Windows hostname restrictions.

Back

Next

ASP.NET Service

Create Service



Service Name

Administrator password

Confirm password

Domain

Not in domain



Git repository

Hostname template

Optional

ASP.NET Application Service

ASP.NET application will be installed onto one IISWeb Server

Service Name: Enter a desired name for a service. Just A-Z, a-z, 0-9, dash and underline are allowed.

Passwords: Windows requires strong password for service administration. Your password should have at least one letter in each register, a number and a special character. Password length should be a minimum of 7 characters.

Domain: Service can be joined to the Active Directory domain. If you want to create an AD domain create the AD Service first.

Git repository: URL of a git repository with the application you want to deploy.

Hostname template: For your convenience all instance hostnames can be named in the same way. Enter a name and use # character for incrementation. For example, host# turns into host1, host2, etc. Please follow Windows hostname restrictions.

Back

Next

ASP.NET Farm Service

Create Service



Service Name

Administrator password

Confirm password

Domain

Not in domain



Git repository

Instance Count

2

Load Balancer port

80

Hostname template

Optional

ASP.NET Application Web Farm Service

The ASP.NET application will be installed on a number of IIS Web Servers, and load balancing will be configured.

Service Name: Enter a desired name for a service. Just A-Z, a-z, 0-9, dash and underline are allowed.

Passwords: Windows requires strong password for service administration. Your password should have at least one letter in each register, a number and a special character. Password length should be a minimum of 7 characters.

Domain: Service can be joined to the Active Directory domain. If you want to create an AD domain create the AD Service first.

Git repository: URL of a git repository with the application you want to deploy.

Instance Count: Several instances with ASP.NET application can be created at one time.

Load Balancer port: Specify port number where Load Balancer will be running

Hostname template: For your convenience all instance hostnames can be named in the same way. Enter a name and use # character for incrementation. For example, host# turns into host1, host2, etc. Please follow Windows hostname restrictions.

Back

Next

MS SQL Service

Create Service×

<div>Service Name</div> <div></div>	MS SQL Server Service MS SQL Server
<div>Administrator password</div> <div></div>	Service Name: Enter a desired name for a service. Just A-Z, a-z, 0-9, dash and underline are allowed.
<div>Confirm password</div> <div></div>	Passwords: Windows requires strong password for service administration. Your password should have at least one letter in each register, a number and a special character. Password length should be a minimum of 7 characters.
<div>Domain</div> <div>Not in domain ▼</div>	Domain: Service can be joined to the Active Directory domain. If you want to create an AD domain create the AD Service first.
<div><input checked="" type="checkbox"/> Mixed-mode Authentication</div>	Mixed-mode Authentication: Mixed authentication mode allows the use of Windows credentials but supplements them with local SQL Server user accounts that the administrator may create and maintain within SQL Server. If this mode is on SA password is required
<div>SA Password</div> <div></div>	SA Password: Set system administrator password for the MS SQL Server.
<div>Confirm password</div> <div></div>	Hostname template: For your convenience all instance hostnames can be named in the same way. Enter a name and use # character for incrementation. For example, host# turns into host1, host2, etc. Please follow Windows hostname restrictions.
<div>Hostname template</div> <div>Optional</div>	
<div>BackNext</div>	

SQL Server Failover Cluster

Create Service

Service Name

Administrator password

Confirm password

☐ Active Directory is configured by the System Administrator

Domain

demo.com

☒ Mixed-mode Authentication

SA Password

Confirm password

MS SQL Server Cluster Service

MS SQL Failover Cluster

Service Name: Enter a desired name for a service. Just A-Z, a-z, 0-9, dash and underline are allowed.

Passwords: Windows requires strong password for service administration. Your password should have at least one letter in each register, a number and a special character. Password length should be a minimum of 7 characters.

Domain: Service can be joined to the Active Directory domain. If you want to create an AD domain create the AD Service first.

Mixed-mode Authentication: Mixed authentication mode allows the use of Windows credentials but supplements them with local SQL Server user accounts that the administrator may create and maintain within SQL Server. If this mode is on SA password is required

SA Password: Set system administrator password for the MS SQL Server.

Back

Next

Enter or select values for the following fields:

Service Name	Enter a desired name for a service. Just A-Z, a-z, 0-9, dash and underline are allowed.
Administrator password	Windows requires strong password for service administration. Your password should have at least one letter in each register, a number and a special character. Password length should be a minimum of 7 characters.
Confirm password	Password confirmation is required.
Active Directory is configured by the System Administrator	Enable this option only if you have properly configured rules that will include service to the domain, that already exists in you environment. Once you set this option to true, additional fields will appear. (See information below)
Domain	Service should be joined to the Active Directory domain. Please, create Active Directory Service prototype first.
Mixed-mode Authentication	Mixed authentication mode allows the use of Windows credentials but supplements them with local SQL Server user accounts that the administrator may create and maintain within SQL Server. If this mode is on SA password is required

SA Password	Set system administrator password for the MS SQL Server. Password requirements are the same as previous.
-------------	--

In case pre-configured AD is enabled following fields will appeared:

☒ **Active Directory is configured by the System Administrator**

Active Directory User

Active Directory Password

Confirm password

Domain

Active Directory User Active Directory Password	Specify administrator user credentials to the existent AD domain (to which service service will be join according to your system automation setup)
---	--

Create Service

Cluster Static IP

Cluster Name

Availability Group Name

Availability Group Listener Name

Availability Group Listener IP

SQL User Name

SQL User Password

Confirm password

Instance Count

2

Hostname template

Optional

MS SQL Server Cluster Service

Cluster Static IP: Specify a valid IPv4 fixed IP.

Cluster Name: Specify a name of a cluster. Just A-Z, a-z, 0-9, dash and underline are allowed.

Availability Group Name: Specify a name of an AG. Just A-Z, a-z, 0-9, dash and underline are allowed.

Availability Group Listener Name: Specify a name of an AG Listener . Just A-Z, a-z, 0-9, dash and underline are allowed.

Availability Group Listener IP: Specify a valid IPv4 fixed IP.

SQL User Name: User name that will be created to manage cluster instances.

SQL User Password: User password that will be created to manage cluster instances.

Instance Count: Microsoft SQL Failover Cluster includes up to 5 instances.

Hostname template: For your convenience all instance hostnames can be named in the same way. Enter a name and use # character for incrementation. For example, host# turns into host1, host2, etc. Please follow Windows hostname restrictions.

Back

Next

Create Service

Nodes

Node	Sync	Primary
node1	<input checked="" type="checkbox"/>	<input checked="" type="radio"/>
node2	<input checked="" type="checkbox"/>	<input type="radio"/>

Add node
Remove node

Database list

MS SQL Server Cluster Service

Nodes: Configure cluster instances. Cluster node quantity can be set with 'Add' and 'Remove' buttons. Configure Sync mode by enabling corresponding checkbox. All other nodes will be in Async mode. Just 2 nodes are allowed to be Sync. Also one Master node need to be selected. SQL Failover cluster has limit of 5 instances.

Database list: Specify names for new databases which will be created as part of service installation. Here should come comma-separated list of database names, where each name has the following syntax: first symbol should be latin letter or underscore; subsequent symbols can be latin letter, numeric, underscore, at sign, number sign or dollar sign.

Back
Next

Deploying environment

Once all services are prepared you can send environment to deploy. Just press the "Deploy Environment" button.

Project

CURRENT PROJECT
Anastasia

Manage Compute

Overview
Instances
Volumes
Images & Snapshots
Access & Security

Other

Environments

Environment: Demo

environments > environment Demo

Logged in as: anastasia
Settings
Help
Sign Out

Services

+ Create Service
Delete Services
Deploy This Environment

	Name	Type	Status	Last operation	Time updated	Actions
<input type="checkbox"/>	demo.com	Active Directory	Configuring	Service draft created	2013-09-05 01:01:30	Delete Service
<input type="checkbox"/>	IISSdemo	Internet Information Services Web Farm	Configuring	Service draft created	2013-09-05 01:02:19	Delete Service

Displaying 2 items

And you'll see a message about successful start of deploying your services in Openstack. Since now all you have to do is just wait for a little bit while Murano installing and configuring your services.

Environment: Demo
environments > environment Demo

Success: Deploy started

Services

Name	Type	Status	Last operation	Time updated	Actions
demo.com	Active Directory	Deploy in progress	Service draft created	2013-09-05 01:01:30	
IISdemo	Internet Information Services Web Farm	Deploy in progress	Service draft created	2013-09-05 01:02:19	

Displaying 2 items

You can monitor deploying process. Just go to the Log tab on service detailed page, where you can get by clicking on the service name.

Environment: Demo
environments > environment Demo

Services

Name	Type	Status	Last operation	Time updated	Actions
demo.com	Active Directory	Ready	Primary Domain Controller created	2013-09-05 21:44:01	Delete Service
IISdemo	Internet Information Services Web Farm	Ready	Unit rnhfth@0713 (IISdemo_instance_0) has joined domain demo.com	2013-09-05 21:47:24	Delete Service

Displaying 2 items

And now you can see installation progress.

Service Detail: demo.com
environments > environment Demo > service demo.com

Service Logs

- 2013-09-05 21:25:41 - Creating instance ad (demo.com_instance_0)
- 2013-09-05 21:28:16 - Instance ad (demo.com_instance_0) created
- 2013-09-05 21:28:19 - Creating Primary Domain Controller on unit ad (demo.com_instance_0)
- 2013-09-05 21:44:01 - Primary Domain Controller created

As long as installation and configuration are in progress, environment is in *Deploying* state. Depending on how many services you are deploying or how many nodes in your cluster, process of spawning instances, installation and post installation settings takes from 10 minutes up to one hour.



Environments

Logged in as: anastasia [Settings](#) [Help](#) [Sign Out](#)

Environments + Create Environment

<input type="checkbox"/>	Name	Status	Actions
<input checked="" type="checkbox"/>	Demo	Deploy in progress	Services More ▾
<input type="checkbox"/>	Virtual DC	Ready to configure	Services More ▾

Displaying 2 items

If installation process finished without any errors, environment changes its status to *Ready*:



Environments

Logged in as: anastasia [Settings](#) [Help](#) [Sign Out](#)

Environments + Create Environment

<input type="checkbox"/>	Name	Status	Actions
<input checked="" type="checkbox"/>	Demo	Ready	Services More ▾

Displaying 1 item

Working with deployed environment

Congratulations! After some time waiting you are able to operate with the services. To get information about installed services, navigate to service detailed page. To do that click on the environment name and then on the name of the service you want to know about.



The screenshot shows the OpenStack Murano dashboard interface. On the left is a sidebar with the OpenStack logo and navigation links for Project, Admin, and Manage Compute (Overview, Instances, Volumes, Images & Snapshots, Access & Security). The main content area is titled 'Service Detail: IISdemo' and shows the service's details. The service is named 'IISdemo' and has a status of 'Ready'. It is a 'Web Farm' type. The domain is 'demo.com' and the load balancer URI is '80'. The service has two units: 'Service unit1' and 'Service unit2'. Each unit has a hostname and a service instance name. The service instance names are hyperlinks that lead to the instance details page.

Service Detail: IISdemo
[environments](#) > [environment Demo](#) > service IISdemo

Service | Logs

Service Details

Info

Name
IISdemo

ID
83e46d472d3d49199763bec4b4ef6e44

Type
Internet Information Services Web Farm

Status
Ready

Domain
demo.com

Load Balancer URI
80

Service unit1

Hostname
mhjfh8h0713

Service instance name
[ef489cd02b66d4abd8b251fedcd0d350f.mhjfh8h0713](#)

Service unit2

Hostname
llchhi8h0774

Service instance name
[ef489cd02b66d4abd8b251fedcd0d350f.llchhi8h0774](#)

172.18.79.30/horizon/project/murano/environments

Now you are seeing general information about the service in terms of Murano Environment. To get information about the instance in Openstack terms follow the link on service instance name.

Service unit1

Hostname
IIS1

Service instance name
[e98a933da40ee4dc096b7f7d7be4fa3de.IIS1](#)

You can login to the virtual machine directly from the horizon (if your Openstack installation allows you) or by RDP protocol.



There are more things you can do with Murano Environment:

- Add new services and deploy it again;
- Delete outdated and unnecessary environments or services;
- Browse deployment history and service installation logs;
- Rename your environment.

Redeploying Murano Environment

Murano gives an opportunity to supplement already deployed environment. Thus if you already deployed the Active Directory service and want to add any other services just create desired service prototype and click the "Deploy Environment" button. During service prototype creation you can join this service to the existent Active Directory domain.

Deleting

Services as well as environments can be easily deleted.

- To delete an environment go to the environment index page and click "More" -> "Delete Environment" in Actions column of ready to delete environment.



Environment deletion means to kill all services with instances on which they are installed. Instances will be scheduled to delete right after you choose the "Delete Environment" action.

- To delete a service go service list page and click the "Delete Service" button in Actions column. **Note:** If you are deleting service that was already deployed you'll need to *Deploy* the environment again by pressing corresponding button. In case you want to delete service prototype - it has *"Service draft created"* in the *Last operation* column (see the screenshot below) - changes applies right away.



Browsing Murano Deployment

Since Murano Environment can be deployed many times you may want to see the history of its deployments. To do that click the "More-> Show deployments" button on environments index page:

The screenshot shows the OpenStack Murano dashboard interface. On the left is a sidebar with the OpenStack logo and navigation links for 'Project' and 'Admin'. Under 'Project', there's a 'CURRENT PROJECT' section for 'Anastasia' and a 'Manage Compute' section with links to Overview, Instances, Volumes, Images & Snapshots, and Access & Security. Under 'Other', there's a link to 'Environments'. The main content area is titled 'Environments' and shows a table with one environment named 'Demo' in a 'Ready' status. A 'Create Environment' button is in the top right. An 'Actions' dropdown menu is open for the 'Demo' environment, showing options: 'Deploy Environment', 'Edit Environment', 'Delete Environment', and 'Show Deployments'.

Name	Status	Actions
Demo	Ready	Services More Deploy Environment Edit Environment Delete Environment Show Deployments

From this page it's easy to see how many times and when Murano Environment was deployed:

The screenshot shows the 'Deployment logs' page in the OpenStack Murano dashboard. The breadcrumb trail is 'environments > Demo deployments'. It displays a table of deployments with columns for 'Time Started', 'Time Finished', 'Status', and 'Actions'. Two successful deployments are listed. The 'Actions' column contains 'Show Details' buttons for each deployment.

Time Started	Time Finished	Status	Actions
2013-09-06 21:29:35	2013-09-06 21:45:09	Successful	Show Details
2013-09-06 20:23:52	2013-09-06 20:31:14	Successful	Show Details

For each deployment you can get a detailed information by clicking the "Show Details" button. You always can go back to any level using navigation string at the page header. From here you can observe what services were installed during deployment:

The screenshot shows the 'Deployment information' page, specifically the 'Logs' tab. The breadcrumb trail is 'environments > Demo deployments > deployment at 2013-09-06 21:29:35'. It displays a table of deployed services with columns for 'Name' and 'Type'. Two services are listed: 'IISdemo' (Internet Information Services Web Farm) and 'demo.com' (Active Directory).

Name	Type
IISdemo	Internet Information Services Web Farm
demo.com	Active Directory

Also deployment logs are available at the "Logs" tab:

The screenshot shows the Murano dashboard interface. On the left is a sidebar with the OpenStack logo and navigation links: Project, Admin, CURRENT PROJECT Anastasia, Manage Compute (Overview, Instances, Volumes, Images & Snapshots, Access & Security), Other (Environments). The main content area is titled 'Deployment information' and shows the user is logged in as 'anastasia'. Below this, there are tabs for 'Configuration' and 'Logs'. The 'Logs' tab is active, displaying a list of deployment events for a deployment at 2013-09-06 21:29:35. The events include: Deployment scheduled, Creating instance IIS1 (IISdemo_instance_0), Creating instance IIS2 (IISdemo_instance_1), Creating instance ad (demo.com_instance_0), Instance IIS1 (IISdemo_instance_0) created, Instance IIS2 (IISdemo_instance_1) created, Instance ad (demo.com_instance_0) created, Creating Primary Domain Controller on unit ad (demo.com_instance_0), Primary Domain Controller created, and Deployment finished.

Renaming Murano Environment

It's possible to change a name of your environment: just click the "More-> Edit Environment" button on environment index page:

The screenshot shows the 'Environments' page in the Murano dashboard. The user is logged in as 'anastasia'. The page displays a table with one environment named 'Demo' with a status of 'Ready'. To the right of the table, there is a 'More' button. A dropdown menu is open, showing options: Deploy Environment, Edit Environment (highlighted), Delete Environment, and Show Deployments. The 'Edit Environment' option is the one to be clicked to rename the environment.

Environment name is not involved in service creation process so you can use spaces and any other characters you want.

The screenshot shows the 'Edit Environment' dialog box open over the 'Environments' page. The dialog has a title bar 'Edit Environment' and a close button. It contains a tab 'Environment Info' and a text input field for 'Name' with the value 'Demo'. To the right of the input field, there is a message: 'From here you can edit the environment details.' At the bottom of the dialog, there are 'Cancel' and 'Save' buttons.

Troubleshooting

How to debug OpenStack Heat?

If you can execute Heat command via console interface - all good. It is the most simple way to check Heat state on the node - just execute CLI command 'heat list'. See more information about Heat in openstack wiki page [<https://wiki.openstack.org/wiki/Heat/TroubleShooting>]

If 'heat list' returns 503 error

It means that OpenStack Heat configuration files contain incorrect credentials. Need to set 'user' = 'heat' and change passwords 'verybadpass' in all configuration files from directory /etc/heat/

If 'heat list' hangs up

Sometimes you can see that 'heat list' hangs up. The root of this problem - connection to the rabbitMQ.

How I can connect to LoadBalancer instance in Server Farms?

First of all you should have KeyPair file 'murano-lb-key'. You can create this file using commands

```
nova keypair-add murano-lb-key > murano-lb-key.priv
chmod 600 murano-lb-key.priv
```

And after that server farms need to be created with this KeyPair. The second step is to 'how to connect to VM with LoadBalancer':

```
ssh -i murano-lb-key.priv root@10.0.0.3
```

Murano dashboard can not connect to Murano API. How I can fix it?

This problem has two ways to fix: Add string

```
MURANO_API_URL='http://localhost:8082'
```

to the /etc/openstack-dashboard/local_settings (or /etc/openstack-dashboard/local_settings.py - it depends on OpenStack configuration) and after that web server restart is needed. Add keystone endpoints for Murano API

```
keystone service-create --name muranoapi --type murano --description "Murano-API S
keystone endpoint-create --region RegionOne --service-id
--publicurl http://localhost:8082 --internalurl http://localhost:8082 --adminurl h
```

Murano API Service does not work on CentOS 6.x. WebUI can not connect to this service. How to fix this?

The problem in pip lib routes. Need to upgrade this lib and restart Murano API:

```
python-pip install routes --upgrade
initctl stop murano-api
initctl start murano-api
```

Error 'Unexpected state' during the deployment of Web Farms. What the problem?

Sometimes we can see in deployments logs:

```
2013-08-06 09:10:07 - Unable to deploy instance ipkrmhk0vzq4b6 (asp-farm_instance_)
2013-08-06 09:10:07 - Unable to create a Server Farm load balancer on unit ipkrmhk
```

The root of this problem is incorrect configuration - Heat can not create Load Balancer instance. Please, remember that you should have admin access for the project in OpenStack to deploy LoadBalancer and also, you should have KeyPair with default name 'murano-lb-key'.

Error in Murano API logs 'No module named helpers.token_sanitizer'

This pip version problem. Need to install pip 1.4 and after that reinstall murano-client, murano-common and murano-api.