vOneCloud Documentation

Release 1.8.0

OpenNebula Systems

October 09, 2015

1 Release Notes vOneCloud 1.8.0 **1**
1.1 What's New vOneCloud 1.8 . . . 1
1.2 Upgrade . . . 2
1.3 System Requirements . . . 2
1.4 Known Issues and Limitations . . . 4

2 Overview **5**
2.1 Introduction . . . 5
2.2 What Is? . . . 5
2.3 vOneCloud Features . . . 6
2.4 Components . . . 8
2.5 Accounts . . . 10

3 Simple Cloud Deployment **13**
3.1 All About Simplicity . . . 13
3.2 Download and Deploy . . . 13
3.3 Import Existing vCenter . . . 22
3.4 Create a Virtual Datacenter . . . 29
3.5 vOneCloud Interfaces . . . 32

4 Security and Resource Consumption Control **37**
4.1 Introduction . . . 37
4.2 Users, Groups and ACLs . . . 37
4.3 Resource Quotas . . . 39
4.4 Accounting & Monitoring . . . 41
4.5 Showback . . . 42

5 Guest Configuration **45**
5.1 Introduction . . . 45
5.2 Building a Template for Contextualization . . . 45
5.3 Guest Contextualization . . . 48
5.4 vCenter Customization . . . 49

6 Infrastructure Configuration **51**
6.1 Introduction . . . 51
6.2 Add New vCenters, VM Templates and Networks . . . 51
6.3 Hybrid Clouds . . . 56
6.4 Multi VM Applications . . . 64
6.5 Authentication . . . 66
6.6 Resource Pool Confinment . . . 69

7 Appliance Configuration **71**
7.1 Introduction 71
7.2 Control Console 71
7.3 Control Panel 72
7.4 Advanced Customizations 76
7.5 Troubleshooting 77

CHAPTER

ONE

RELEASE NOTES VONECLOUD 1.8.0

1.1 What's New vOneCloud 1.8

vOneCloud 1.8 is powered by OpenNebula Great A'Tuin, and, as such, includes functionality present in OpenNebula 4.14 Great A'Tuin relevant to vOneCloud:

- *Import VMs from public clouds*. vOneCloud 1.8 introduces the management of VMs not launched by vOneCloud in either Amazon EC2, Microsoft Azure and IBM SoftLayer.
- *Keep the VM disks after deletion*. A new attribute, KEEP_DISKS_ON_DONE will instruct vOneCloud on leaving or deleting the VM disks when they finish their lifecycle.

Additionally, vOneCloud 1.8 add new features related with the vCenter driver not yet present in any OpenNebula release:

- *Pagination added to vCenter import dialog*. In large scale deployments, this will aid in the importation of large numbers of VM Templates and Networks.
- *Support for Guest Customization*. At the time of creating or modifying an imported VM Template, it is now possible to associate a Guest Customization profile to configure guest OS parameters such as the computer name, network settings, setting and expiring the administrator/root password, SID change for Windows Operating systems, and so on.
- *Show all the IPs from a VM in Sunstone*. Both for imported and newly created VMs, all the IP addresses reported by the guest OS via the VMware tools are retrieved by vOneCloud and presented in Sunstone.
- *Append name to "one-*" name in vCenter display name*. VM Name as shown in vOneCloud is appended in vCenter VM name for easier VM tracking.
- *Imported resources suffixed with their source cluster name*. This feature aids in the resource identification in the vCenter portal by mapping it to the vOneCloud representation name.
- *Multi cluster VM Template definition*. Create vOneCloud VM Templates that reference more than one vCenter VM Template in different vCenter clusters.

The Control Panel has also been extended in this release:

- *Enable SSH*. To easily allow console access to the vOneCloud appliance.
- *Enable SSL*. To enable the secure access to the vOneCloud appliance web interface (Sunstone) using the SSL protocol.

Additionally, a new documentation category starts with vOneCloud 1.8 to explain advanced customizations to the appliance that are not performed through the Control Panel:

- *vOneCloud rebranding*. hange the logos of the Sunstone interface.

Multiple bugfixes and documentation improvements have been included in this version. vOneCloud 1.8 has been certified with *support for vSphere 5.5 and 6.0*.

The *Automated Upgrade process* implemented by the *Control Panel* will only be available to users with an active support subscription. With this functionality users will be notified when a new vOneCloud release is available for download and they will be able to upgrade the vOneCloud platform with a single click.

1.2 Upgrade

Upgrading to a newer version of vOneCloud is only supported for users with an active support subscription. The upgrade process is carried out in the *Control Panel web interface*.

When a new vOneCloud release is available for download, users with an active support subscription will be notified in the Sunstone interface (in particular, in the Control Panel link), as well as in the main Dashboard area of the Control Panel, and will be able to upgrade with a single click. The Control Panel component will, behind the scenes:

- download the new vOneCloud packages
- install the new vOneCloud packages, keeping the existing configuration
- restart the OpenNebula service, with no downtime whatsoever to the currently running virtual machines.

The Control Panel will display a message after the upgrade is performed, at this moment vOneCloud services would be up and running and updated to the latest version.

1.3 System Requirements

Warning: It is advised to manage one vCenter by only one vOneCloud. Otherwise VMs from both server will clash and poduce errors.

The following components are needed to be present in the infrastructure to implement a cloud infrastructure run by vOneCloud:

Component	Observations
vCenter 5.5 and 6.0	• ESX hosts, VM Templates and Running VMs expected to be managed by vOneCloud needs to be grouped into clusters • The IP or DNS needs to be known, as well as the credentials (username and password) of an admin user. • DRS is not required but it is recommended. vOneCloud does not schedule to the granularity of ESX hosts, and you would need DRS to select the actual ESX host within the cluster. Otherwise the VM will be started in the ESX host associated to the VM Template • All ESX belonging to the same vCenter cluster to be exposed to vOneCloud need to share at least one datastore among them. • VMs that will be instantiated through vOneCloud saved as VMs Templates in vCenter.
ESX 5.5 and 6.0	• With at least 2 GB of free RAM and 1 free CPU • To enable VNC functionality from vOneCloud there are two requirements: 1) the ESX hosts need to be reachable from vOneCloud and 2) the ESX firewall should allow for VNC connections (see the note below)
Guest OS	VMware tools are needed in the guestOS to enable several features (contextualization and networking feedback). Please install VMware Tools (for Windows) or Open Virtual Machine Tools (for *nix) in the guestOS.

Note: To enable VNC functionality for vOneCloud, repeat the following procedure for each ESX:

- In the vSphere client proceed to Home -> Inventory -> Hosts and Clusters
- Select the ESX host, Configuration tab and select Security Profile in the Software category.
- In the Firewall section, select Edit. Enable GDB Server, then click OK.

Make sure that the ESX hosts are reachable from vOneCloud.

---

vOneCloud ships with a default of 2 CPUs and 2 GB of RAM, and as such it has been certified for infrastructures of the following dimensions:

- Up to 4 vCenters
- Up to 40 ESXs managed by each vCenter
- Up to 1.000 VMs in total, each vCenter managing up to 250 VMs
- Up to 100 users, being the concurrent limit 10 users accessing the system simultaneously

---

Note: For infrastructures exceeding the aforementioned limits, we recommend an installation of OpenNebula from scratch on a bare metal server, using the vCenter drivers

---

1.4 Known Issues and Limitations

1.4.1 Known Issues

These known issues will be addressed in future versions of vOneCloud.

Importing vms from EC2 provides a wrong CPU value

CPU consumption shown in vOneCloud is 100x the real value

Error during upgrades if Proxy is configured

There is a problem when upgrading from 1.2.x if proxy is configured that requires a manual intervention. Upgrade normally, and you will see that the start job has failed. Login to the vOneCloud console as explained *here*, and execute the following commands:

```
echo export http_proxy=<yourproxy> > /etc/profile.d/proxy.sh
source /etc/profile.d/proxy.sh
gem install mysql --no-ri --no-rdoc
sudo -u oneadmin onedb upgrade -u oneadmin -p oneadmin -d opennebula
/usr/lib/one/vonecloud-control-center/scripts/opennebula-server.sh restart
```

Found more?

If you find any new issue, please let us know in the Community Questions section of the vOneCloud Support Portal.

1.4.2 Limitations

These limitations will be addressed in future versions of vOneCloud:

Limitation	Description
VM Unsupported Operations	The following operations are only supported from vCenter: - Attach/detach disk to a running VM - Migrate VM to different ESX clusters
No spaces in Clusters	VMware Clusters with space in their names are not supported
No proxy support for SoftLayer	If vOneCloud is running behind a corporate http proxy, the SoftLayer hybrid connectors won't be available
No auth proxy support for Azure	Azure driver only supports proxies without authentication. That is, without username and password.
No FILES support in context	Contextualization in vOneCloud does not support passing files to Virtual Machines
Cannot import "one-" VMs	VMs deployed by another instance of vOneCloud, or machines named with a leading "one-" cannot be imported again
vCenter password length	Cannot be more than 22 characters

If you find any new limitation, feel free to add a feature request in Community - Feature Request section of the vOneCloud Support Portal.

CHAPTER

TWO

OVERVIEW

2.1 Introduction

vOneCloud extends vCenter with cloud features such as provisioning, elasticity, multi-tenancy and multi-vm capabilities. vOneCloud is designed for companies that want to create a self-service cloud environment on top of their VMware infrastructure without having to abandon their investment in VMware and retool the entire stack. vOneCloud leverages advanced features such as vMotion, HA or DRS scheduling provided by the VMware vSphere product family.

This section describes the vOneCloud platform as a whole, and its components, features and roles.

2.2 What Is?

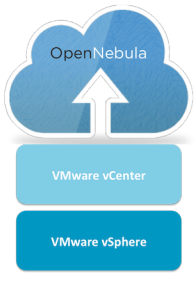
vOneCloud

The Open Replacement for vCloud

vOneCloud is an OpenNebula distribution optimized to work on existing VMware vCenter deployments. It deploys an enterprise-ready OpenNebula cloud just in a few minutes where the infrastructure is managed by already familiar VMware tools, such as vSphere and vCenter Operations Manager, and the provisioning, elasticity, multi-tenancy, elasticity and multi-vm cloud features are offered by OpenNebula. It inherits all the benefits from the open source cloud managment platform, adding an easy to deploy, easy to use aspect due to pre configuration of the OpenNebula install contained within the appliance.

vOneCloud is distributed as a virtual appliance in OVA format for vSphere. It contains all required OpenNebula services within a single CentOS Linux appliance. All components are fully open-source and have been certified to work in enterprise environments, vOneCloud 1.8 includes:

CentOS	7.0
OpenNebula	4.14.1



The following table summarizes the benefits of vOneCloud:

Powerful	Virtual data centers, self-service, datacenter federation, hybrid cloud on VMware environments
Cost Effective	Free, there are no license costs, all componentes are fully open-source software
Flexible	Completely open, customizable and modular, so it can be adapted to your needs
No Lock-in	Platform independent, gradually migrate to other virtualization platforms
Simple	Very easy to install, upgrade, and maintain, with easy-to-use graphical interfaces
Enterprise-ready	Certified, production-ready with commercial support subscriptions and professional services

2.3 vOneCloud Features

vOneCloud leverages the functionality of OpenNebula. The following features come preconfigured and can be used **out-of-the-box** with vOneCloud:

- **Cloud User Interfaces**
  - Simple, clean, intuitive portals for cloud consumers and Virtual Datacenter (VDC) administrators.
- **Cloud Admin Interfaces**
  - SunStone Portal for administrators and advanced users
  - Powerful CLI that resembles typical UNIX commands applications
- **Import Existing Resources**
  - Import existing vCenter VM Templates
  - Import existing vCenter Networks and Distributed vSwitches
  - Import existing running Virtual Machines
- **On-demand Provision of Virtual Data Centers**

  - Dynamic creation of Virtual Data Centers (VDCs) as fully-isolated virtual infrastructure environments where a group of users, under the control of the group administrator, can create and manage compute capacity
  - Placement of VDCs to multiple vCenters
  - Resource Pool Confinment, restrict vOneCloud users to a subset of hardware specified by a Resource Pool
- **Hybrid Cloud**
  - Cloud-bursting of VMs to public clouds
- **Fast Provisioning**
  - Automatic provision of Virtual Machines and Services (Multi-VM applications) from a Template catalog
  - VM Template cloning and editing capabilities to maintain Template catalog
  - Automatic execution and scaling of multi-tiered applications
  - Snapshotting
  - Contextualization capabilities, including the ability to run any script at VM boot time
- **Security and Resource Consumption Control**
  - Resource Quota Management to track and limit computing resource utilization
  - Fine-grained accounting and monitoring
  - Complete isolated VDCs and organizations
  - Fine-grained ACLs and user quotas
  - VM capacity resizing (CPU and Memory)
  - Powerful user, group and role management
  - vCenter Network and Distributed vSwitch support
  - Attach/detach network interfaces funcionality
  - Capacity (CPU and MEMORY) resizing
  - Showback functionality to report resource usage cost
  - VNC connection to VMs, including the ability to set keymap
- **Enterprise Datacenter Component Integration Capabilities**
  - Integration with user management services like Active Directory and LDAP.
  - HTTP Proxy support
- **Reliability, Efficiency and Massive Scalability**
  - Profit from years of testing and production use
  - Be sure that your Cloud Mangement Platform will be up to the task

vOneCloud additionally brings new configuration and upgrade tools:

- **Appliance and Services Configuration**
  - Control Console for vOneCloud appliance configuration
  - Control Panel (Web UI) for vOneCloud services configuration and debugging
- **Smooth Upgrade Process**

  - Automatic upgrade process and notifications through the Control Panel available for users with an active support subscription

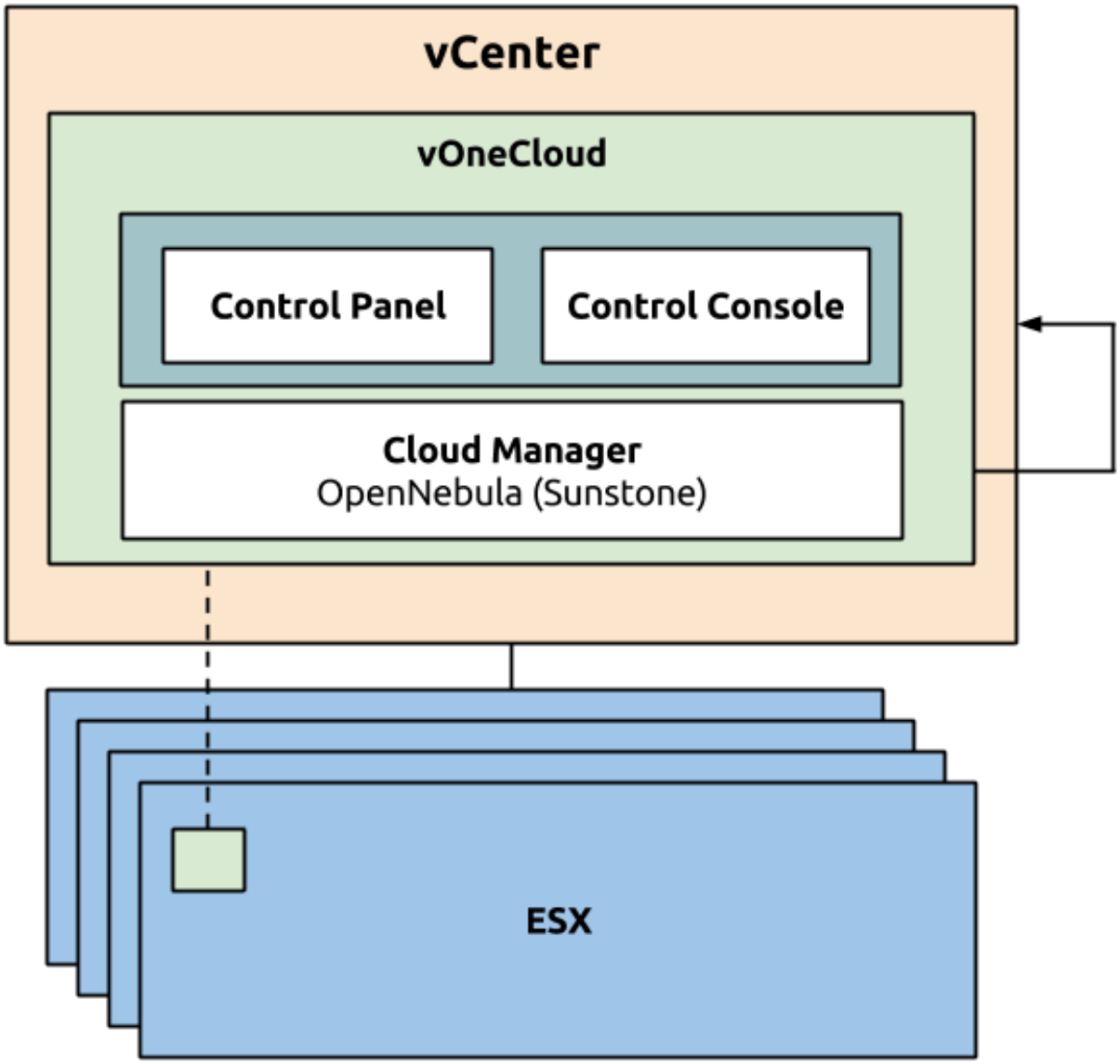
If you feel that there is a particular feature interesting for the general public, feel free to add a feature request in Community - Feature Request section of the vOneCloud Support Portal. vOneCloud can leverage all the functionality that OpenNebula delivers, but some of it needs **additional configuration** steps:

- Centralized Management of Multiple Zones. Federate different datacenters by joining several vOneCloud instances.
- Community Virtual Appliance Marketplace. Create your own marketplace or benefit from community contributions with an online catalog of ready-to-run virtual appliances.
- Broad Commodity and Enterprise Platform Support. Underlying OpenNebula software features an amazingly flexible and plugin oriented architecture that eases the integration with existing datacenter components. Do no reinvent your datacenter, evolve it!
- Virtual & Physical Infrastructure Control. Manage all aspects of your physical (hypervisors, storage backends, etc) & virtualized (VM lifecycle, VM images, virtual networks, etc) from a centralized web interface (Sunstone).

Although the configuration is tailored for vCenter infrastructures, all the power of OpenNebula is contained in vOneCloud and it can be unleashed!

2.4 Components

This diagram reflects the relationship between the components that compose the vOneCloud platform.



2.4.1 vCenter infrastructure

- vOneCloud is an appliance that is executed under vCenter. vOneCloud then leverages this previously set up infrastructure composed of vCenter and ESX nodes.

2.4.2 OpenNebula (Cloud Manager)

- OpenNebula acts as the Cloud Manager of vOneCloud, responsible for managing your virtual vCenter resources and adding a Cloud layer on top of it.
- **Sunstone** is the web-based graphical interface of OpenNebula. It is available at *http://<appliance_ip>*. This interface is at the same time the main administration interface for you cloud infrastructure, and consumer interface for the final users of the cloud.

2.4.3 Control Console and Control Panel

Control Console and Control Panel are two components which have the goal of configuring different aspects of the vOneCloud appliance: network, appliance user accounts, OpenNebula (Sunstone) configuration and services.

- The *Control Console* is a text based wizard accesible through the vCenter console to the vOneCloud appliance and has relevance in the bootstrap process and the configuration of the appliance
- The *Control Panel* is a slick web interface and is oriented to the configuration of the vOneCloud services as well as used to update to a newer version of vOneCloud.

2.5 Accounts

The vOneCloud platform ships with several pre-created user accounts which will be described in this section:

Account	Interface	Role	Description
root	linux	Appliance administrator	This user can log into the appliance (local login, no SSH).
oneadmin	vOneCloud Control Panel	vOneCloud Appliance administrator	Used to configure several aspects of the vOneCloud Appliance infrastructure: OpenNebula services, automatic upgrades, and drivers configuration (hybrid drivers and Active Directory integration).
CloudAdmin	OpenNebula (Sunstone)	Cloud Administrator	Cloud Administrator. Run any task in OpenNebula, including creating other users.

Different *cloud roles* can be used in order to offer and consume cloud provisioning services in Sunstone (vOneCloud Web UI). These roles can be defined through Sunstone, and in particular CloudAdmin comes preconfigured as the Cloud Administrator.

2.5.1 *root* linux account

vOneCloud runs on top of Linux (in particular *CentOS 7 <http://www.centos.org/>*), therefore the administrators of the vOneCloud appliance should be able to have console access to the appliance. The appliance comes with a *root* account with an undefined password. This password **must** be set during the first boot of the appliance. The *vOneCloud Control Console* will prompt the administrator for a new root password.

Please note that ssh acccess to the root account is disabled by default in the appliance, the only possible way of logging in, is to log in using an alternate TTY in the vCenter console of the vOneCloud appliance and logging in.

Note: Console access to the appliance is not required by vOneCloud. Use it only under special circumstances. If you are a user with an active support subscription, make sure any changes applied in the appliance are supported by the vOneCloud support.

2.5.2 *oneadmin* account

The main use of this account is to access the vOneCloud Control Panel (http://<appliance_ip>:8000). Only this account will have access to the Control Panel, no other user will be allowed to log in.

However, the *oneadmin* account is also a valid Sunstone account, but we **strongly recommend not to use this account to access the Sunstone Web UI**, relying instead in the pre-existing *CloudAdmin* account (see below).

The *oneadmin* account password is set by the admin user during the initial configuration of the vOneCloud Control Console. The **password can only be changed in the vOneCloud Control Console**. After changing it the user **must** restart the OpenNebula service in the *vOneCloud Control Panel*.

2.5.3 *CloudAdmin* OpenNebula (Sunstone) account

This account is used to log into Sunstone. It is a Cloud Administrator account, capable of running any task within OpenNebula, however, since this account cannot log into the vOneCloud Control Panel, it cannot control Appliance infrastructure, only the virtual resources.

This account should also be used to create other accounts within Sunstone, either with the same level of privileges (by placing a new account in the *oneadmin* group) or final user without admin privileges. These final users can either be *VDCadmins or cloud consumers*.

The default password for this account is *CloudAdmin* (just like the username). Make sure you change the password within Sunstone once you log in.

CHAPTER

THREE

SIMPLE CLOUD DEPLOYMENT

3.1 All About Simplicity

vOneCloud is preconfigured to be plugged into your existing vCenter infrastructure and quickly start using its cloud features. vOneCloud is the perfect choice for companies that want to create a self-service cloud environment on top of their VMware infrastructure without having to abandon their investment in VMware and retool the entire stack.

Simple to Use	Simple graphical interfaces for cloud consumers, and VDC and cloud administrators.
Simple to Update	New versions can be easily installed with no downtime of the virtual workload.
Simple to Adopt	Add cloud features, do not interfere in existing VMware procedures and workflows.
Simple to Install	CentOS appliance deployable through vSphere, able to import your system

This guide will guide through all the needed steps to deploy vOneCloud and prepare your new cloud to provision your end users.

3.2 Download and Deploy

Download links:

- Download

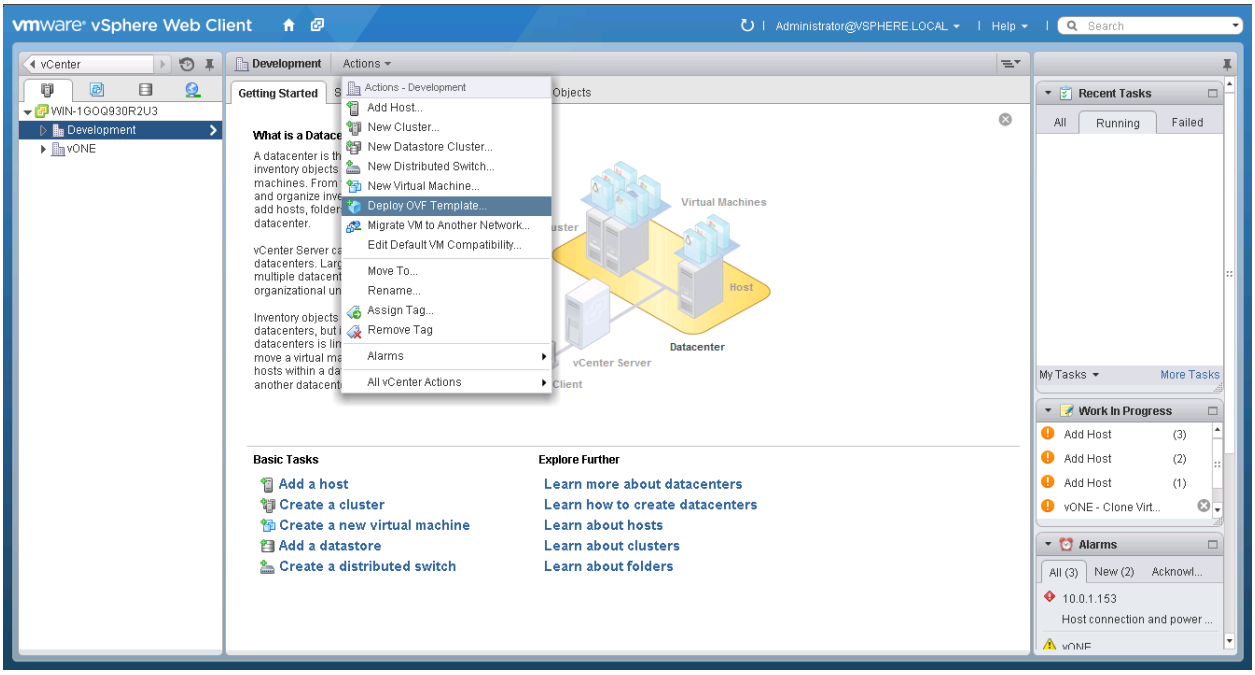
You can import this OVA appliance to your vCenter infrastructure. It is based on CentOS 7 and has the VMware tools enabled.

The appliance requirements are kept to a strict minimum so it can be executed in any vCenter installation. However, before deploying it, please read the *system requirements*.

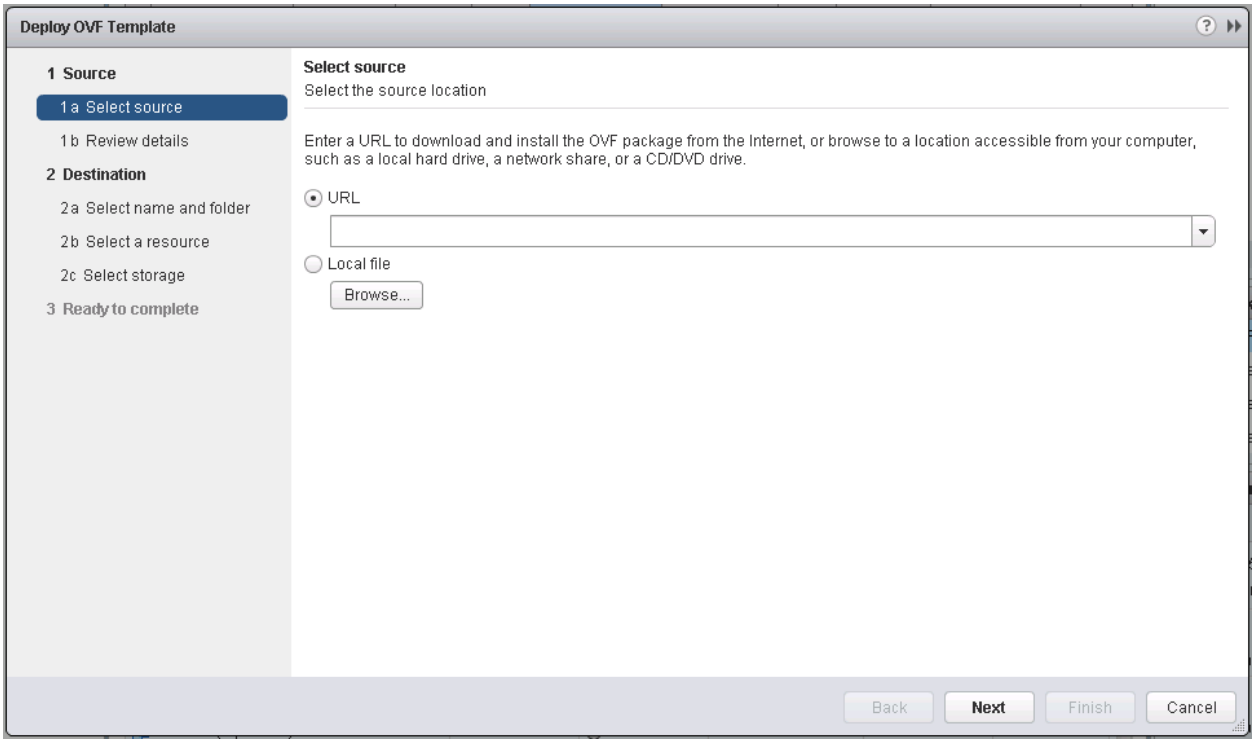
Follow the next steps to deploy a fully functional vOneCloud:

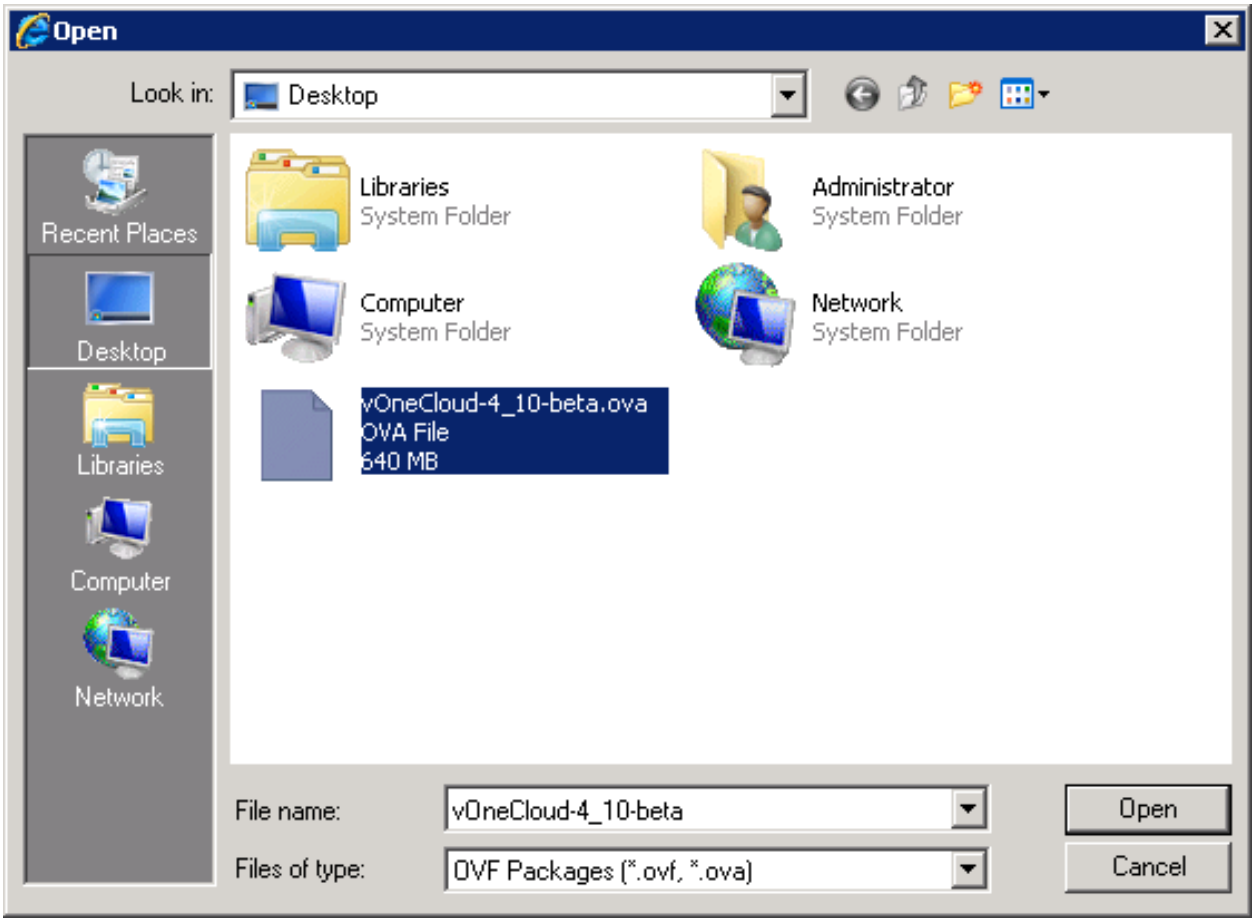
3.2.1 Step 1. Deploying the OVA

Login to your vCenter installation and select the appropriate datacenter and cluster, where you want to deploy the appliance. Select the `Deploy OVF Template`.

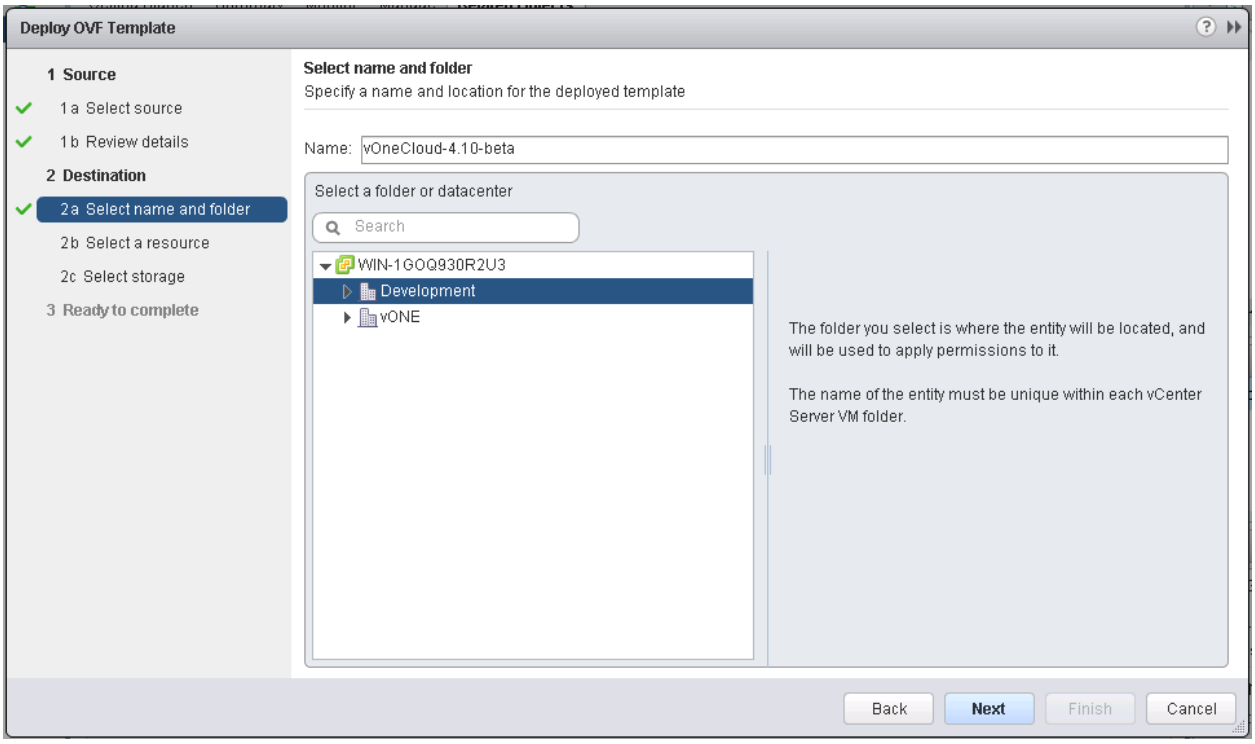


You have the option now to input the URL of the appliance (you can find it at the top of this page), or if you have previously downloaded it, you can simply browse to the download path as such:

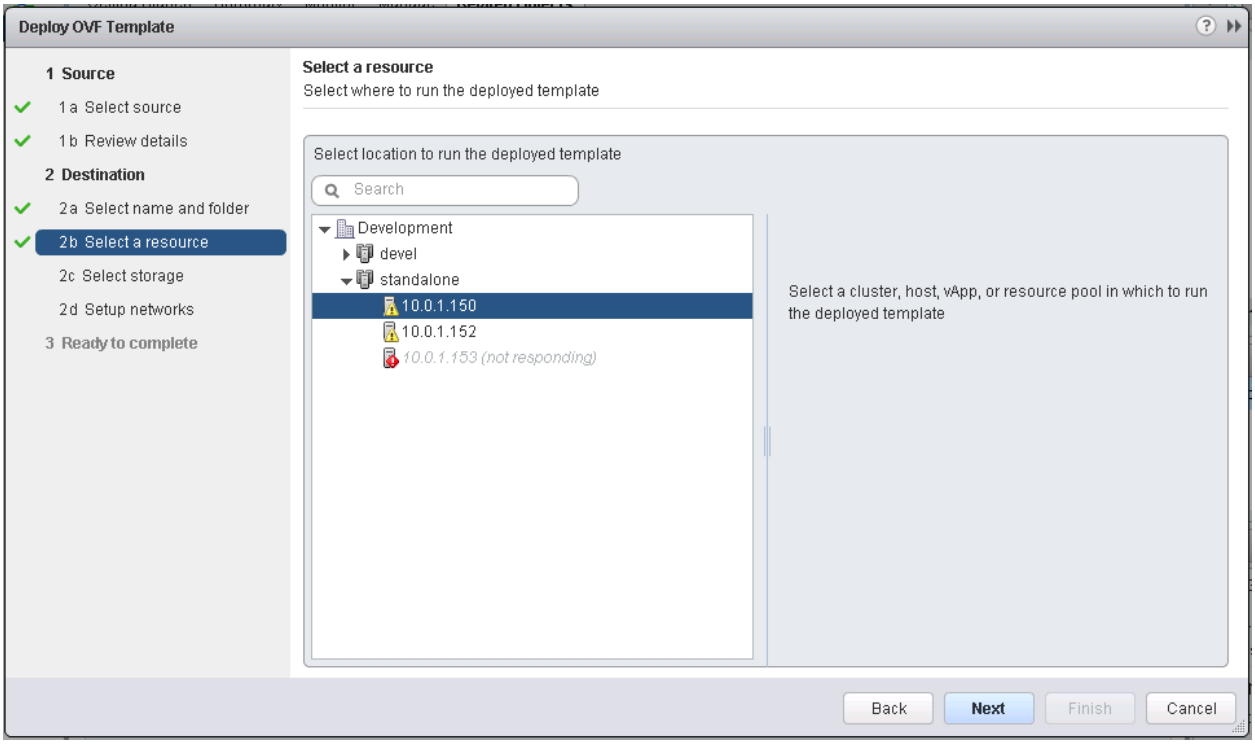




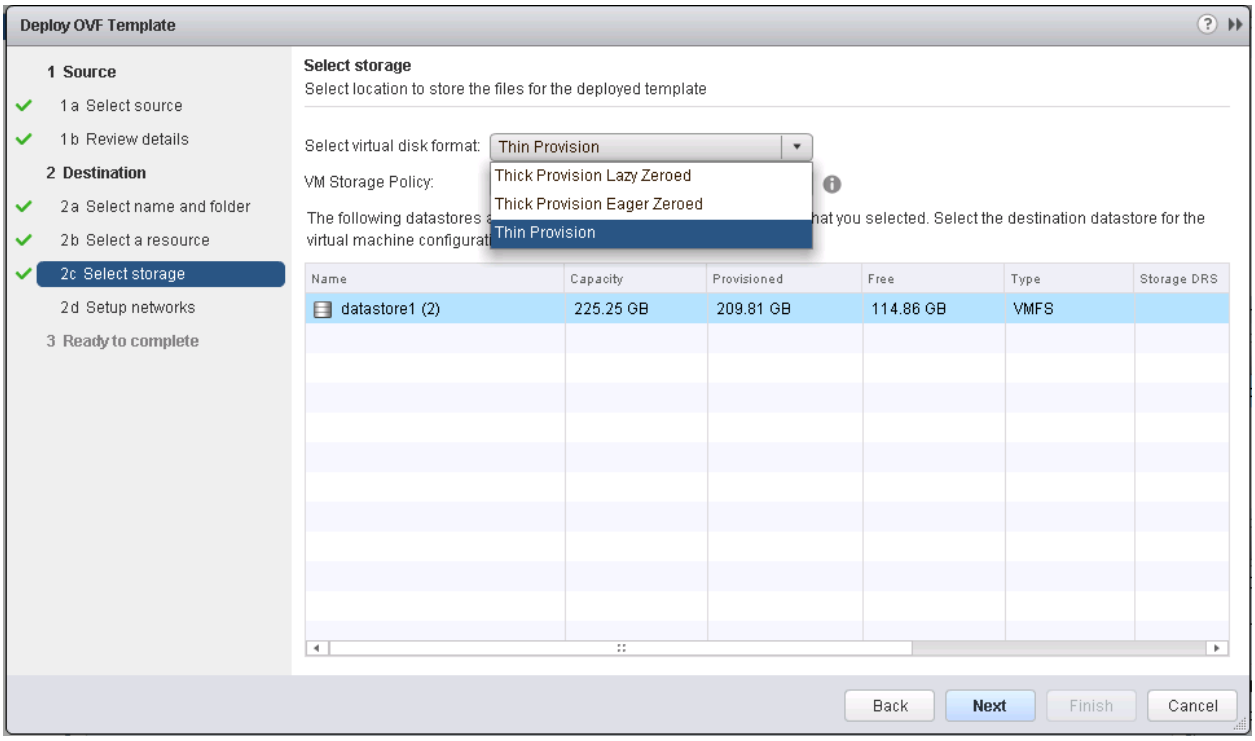
Select the name and folder:



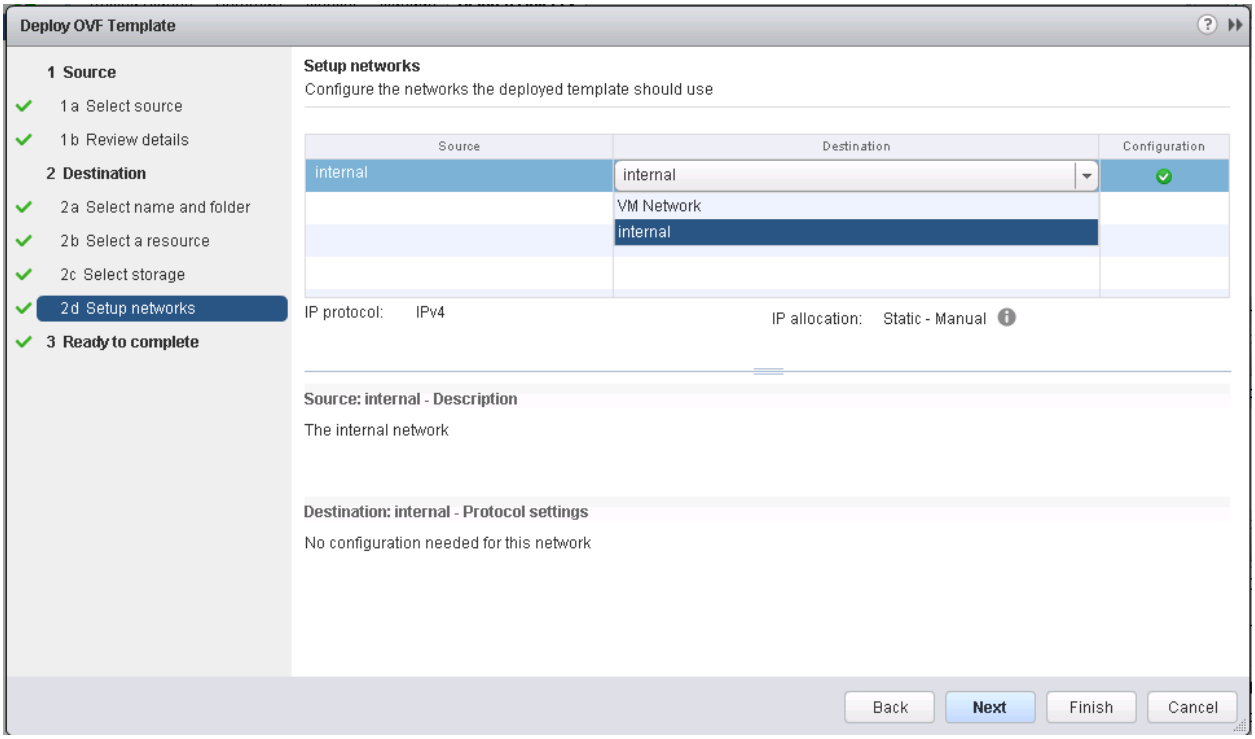
Select a resource to run the appliance:



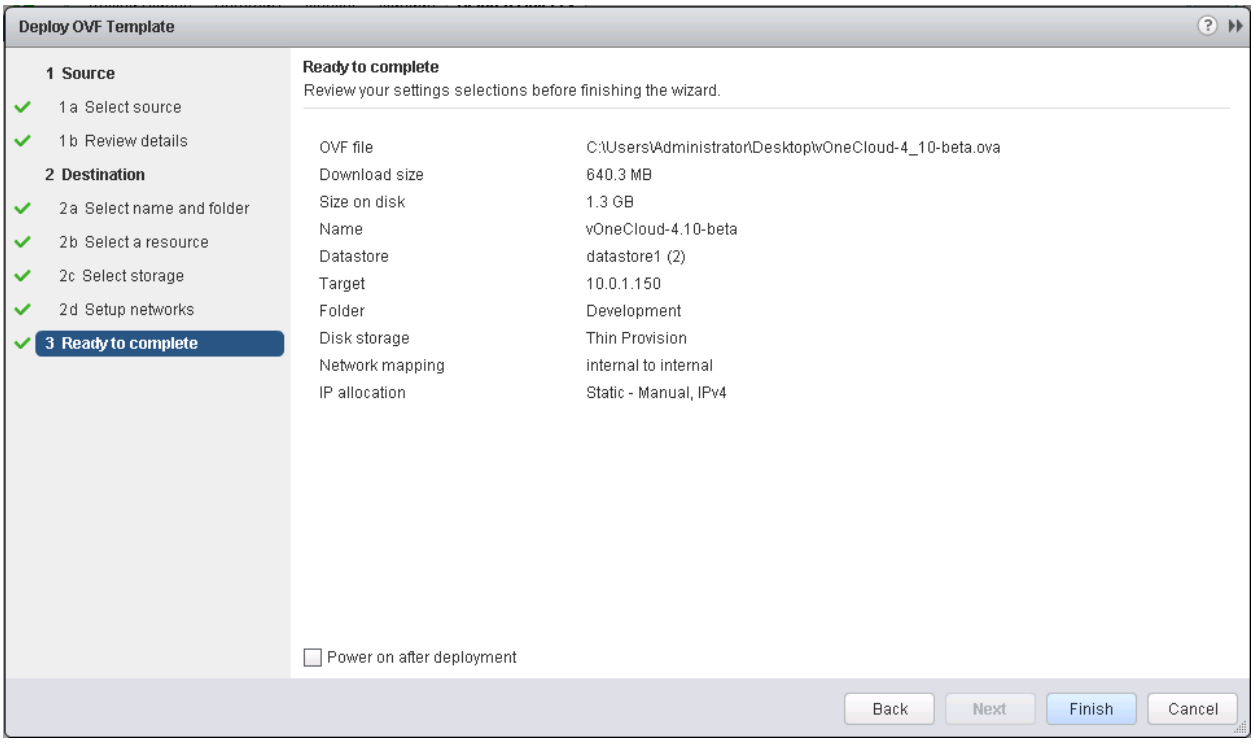
Select the datastore:



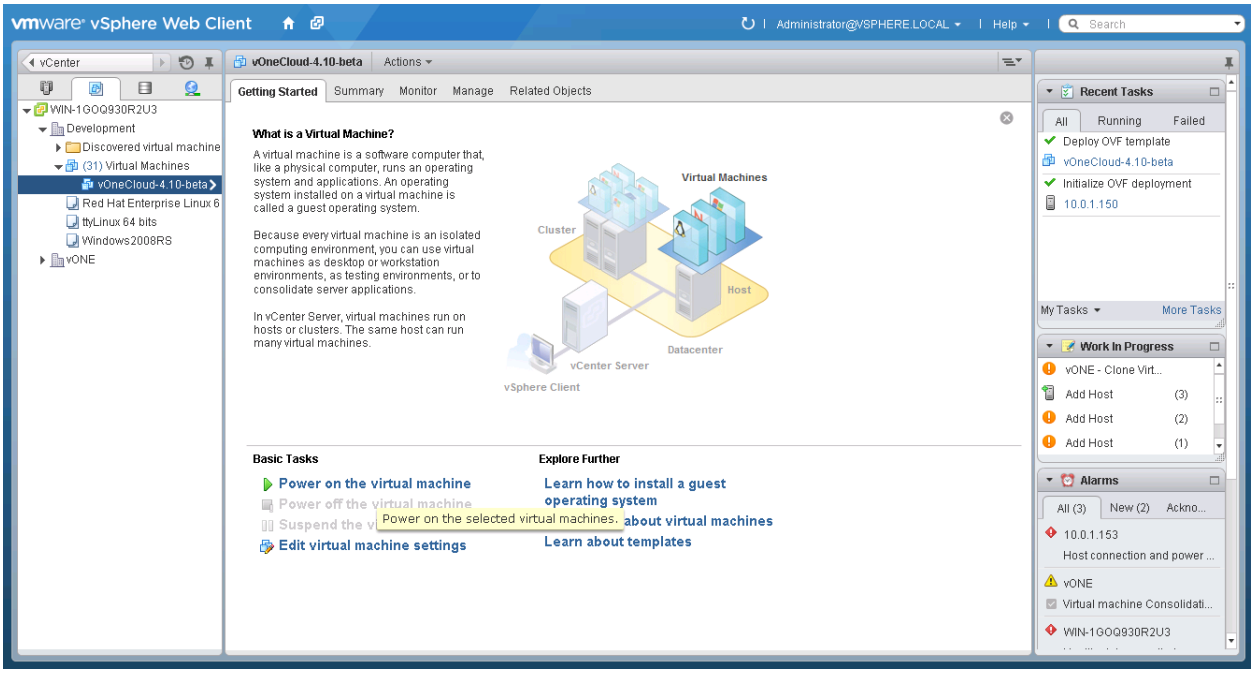
Select the Network. You will need to choose a network that has access to the ESX hosts.



Review the settings selection and click finish. Wait for the Virtual Machine to appear in the cluster.



Now you can power on the Virtual Machine (to edit settings before, *read this section*):



3.2.2 Step 2. vOneCloud Control Console - Initial Configuration

When the VM boots up you will see in the vCenter console in vCenter the *vOneCloud Control Console*, showing this wizard:

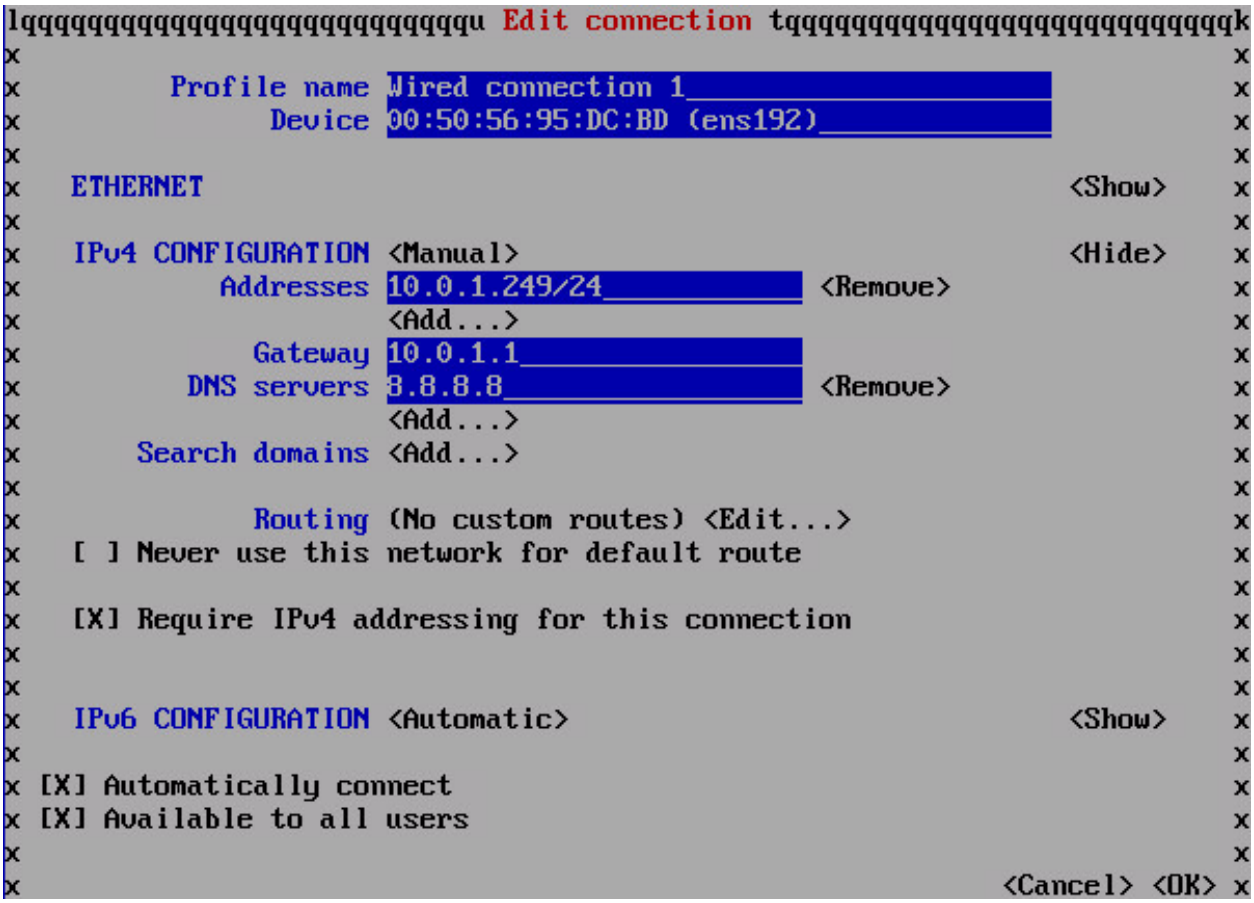
In this wizard you need to **configure the network**. If you are using DHCP you can simply skip to the next item.

If you are using a static network configuration, answer yes and you will need to use a ncurses interface to:

- "Edit a connection"
- Select "Wirect connection 1"
- Change IPv4 CONFIGURATION from <Automatic> to <Manual> and select "Show"

- Input the desired IP address/24 in Addresses
- Input Gateway and DNS Servers
- Select OK and then quit the dialog.

An example of static network configuration on the available network interface (see *Editing the vOneCloud Appliance* for information on how to add new interfaces to vOneCloud) on the 10.0.1.x class C network, with a gateway in 10.0.1.1 and using 8.8.8.8 as the DNS server:



Next, you can **configure the proxy** if your network topology requires a proxy to access the internet. However please note that it's absolutely fine to use vOneCloud without any internet access at all, as you will be able to do most of the things, except for automatic upgrades and hybrid cloud access.

Afterwards you need to define a **root password.** You won't be using this very often, so write it down somewhere safe. It's your master password to the appliance.

The next item is the **oneadmin account password**. You will only need this to login to the vOneCloud Control Panel, a web-based configuration interface we will see very shortly. Check the *Accounts section* to learn more about vOneCloud roles and users.

We have now finished the vOneCloud Control Console initial configuration wizard. As the wizard itself will point out now you can open the vOneCloud Control Panel by pointing your browser to *http://<appliance_ip>:8000* and using the *oneadmin* account and password just chosen.

3.2.3 Step 3. vOneCloud Control Panel - Manage Services

The *vOneCloud Control Panel* will allow the administrator to:

- Check for new vOneCloud versions and manage upgrades.
- Configure Active Directory / LDAP integration and hybrid cloud drivers: Amazon EC2, Windows Azure and IBM SoftLayer.
- Start the OpenNebula services
- Manage automatic upgrades.

Click on the configuration icon if you need to configure one of the supported options. Keep in mind that you can run this configuration at any moment. We recommend to start inspecting vOneCloud's functionality before delving into advanced configuration options like the aforementioned ones.

After clicking on the Start button, proceed to log in to Sunstone (OpenNebula's frontend) by opening: *http://<appliance_ip>* and using the default login *CloudAdmin* / *CloudAdmin* user and password.

Note: There is a guide available that documents the configuration interfaces of the appliance *here*.

3.2.4 Step 4. Enjoy the Out-of-the-Box Features

After opening the Sunstone interface (*http://<appliance_ip>* with *CloudAdmin* / *CloudAdmin* user and password) you are now ready to enjoy the *out-of-the-box features* of vOneCloud!

Move on to the *next section* to start using your cloud by importing your vCenter infrastructure.

3.2.5 Login to the Appliance

Warning: If you make **any** changes to OpenNebula configuration files under `/etc/one` please note that they **will** be either discarded in the next upgrade, or overwritten by vOneCloud Control Center. Keep in mind that only those features configurable in Sunstone or in vOneCloud Control Console and Control Panel are officially supported. Any other customizations are not supported by vOneCloud Support.

All the functionality you need to run your vOneCloud can be accessed via Sunstone, and all the support configuration parameters are available either in the *vOneCloud Control Console* or in the *vOneCloud Control Panel*.

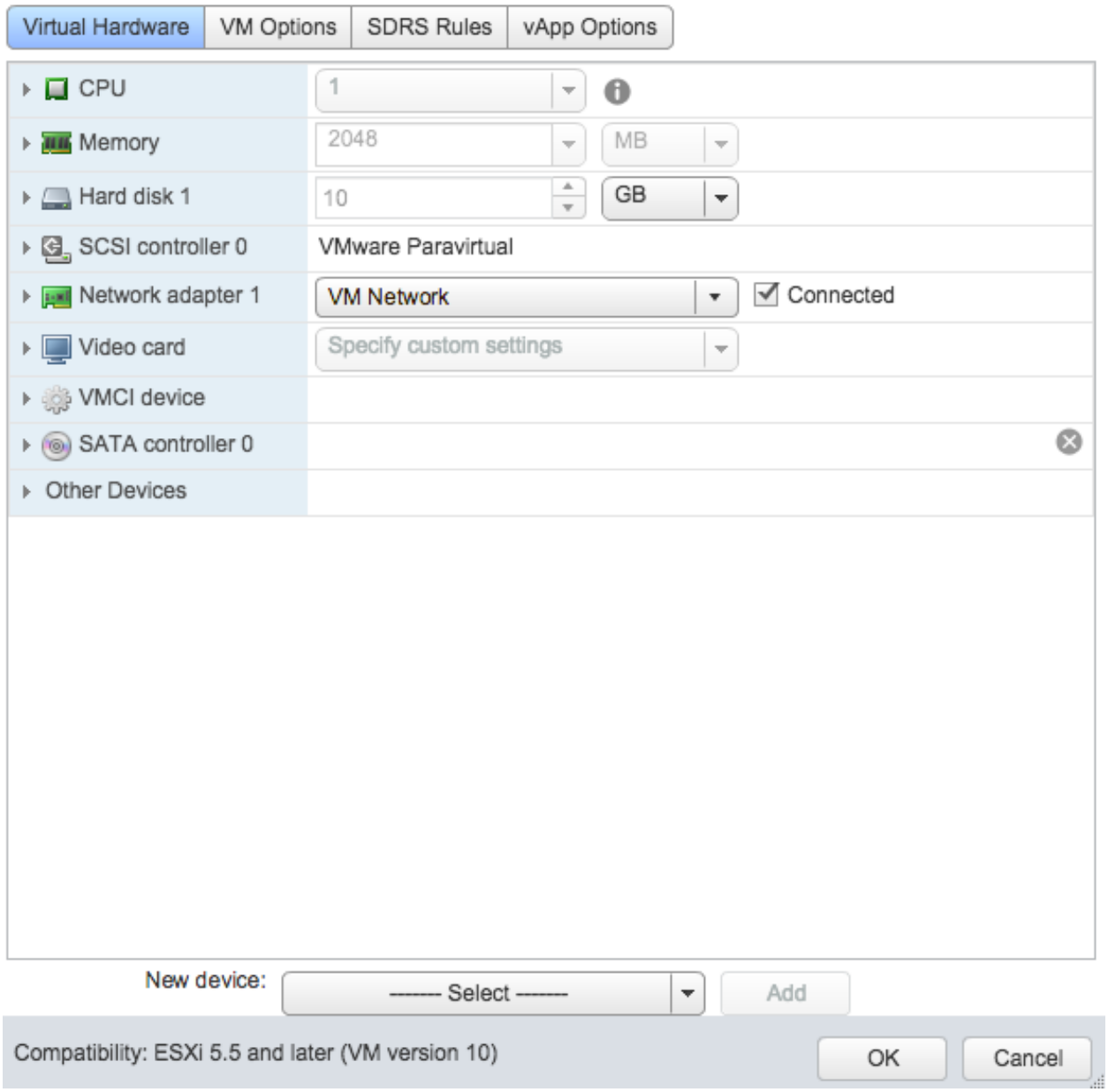
To access the *vOneCloud command line interface*, first *enable SSH* and ssh to the host using the *root* account and password. In OS X and Linux environments, simply use *ssh* to log into the root account of vOneCloud's IP. For Windows environments you can use software like PuTTY or even SFTP clients like WinSCP if you are simply uploading or downloading-modifying-uploading files in order to make customization as documented by the *Advanced Customizations* section.

Alternatively, open the vCenter console of the vOneCloud Virtual Machine appliance and change the tty (Ctrl + Alt + F2). Afterwards, log in with the *root* account and the password you used in the *initial configuration*, and switch to the *oneadmin* user.

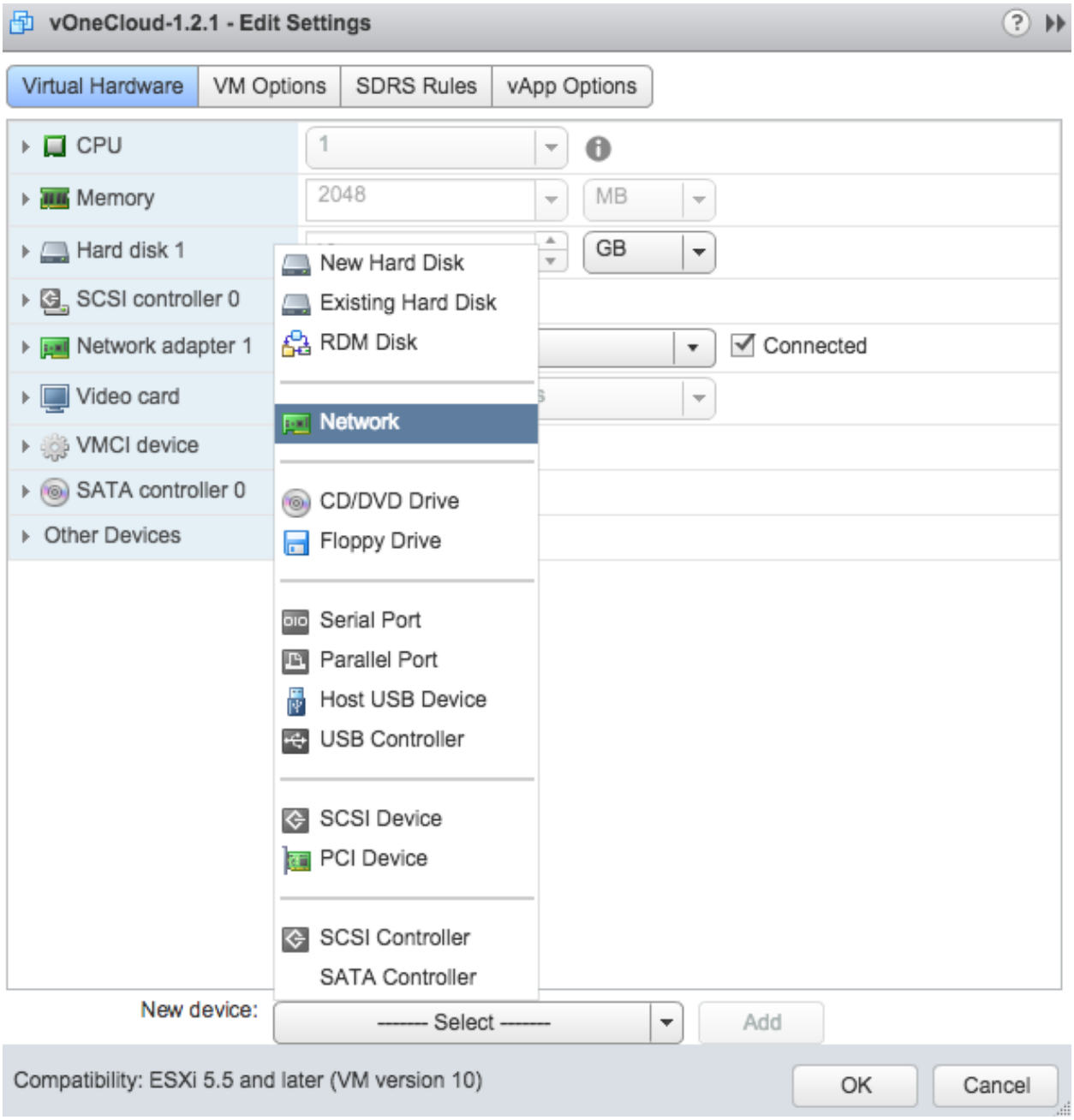
3.2.6 Editing the vOneCloud Appliance

After importing the vOneCloud OVA, and before powering it on, the vOneCloud Virtual Machine can be edited to, for instance, add a new network interface, increase the amount of RAM, the available CPUs for performance, etc.

In order to achieve this, please right click on the vOneCloud VM, and select Edit Settings. The next dialog should pop up:



If you want for instance to add a new network interface, select Network from the dropdown in New device (at the botton of the dialog):



3.3 Import Existing vCenter

Importing a vCenter infrastructure into vOneCloud can be carried out easily through the Sunstone Web UI. Follow the next steps to import an existing vCenter as well as any already defined VM Template and Networks.

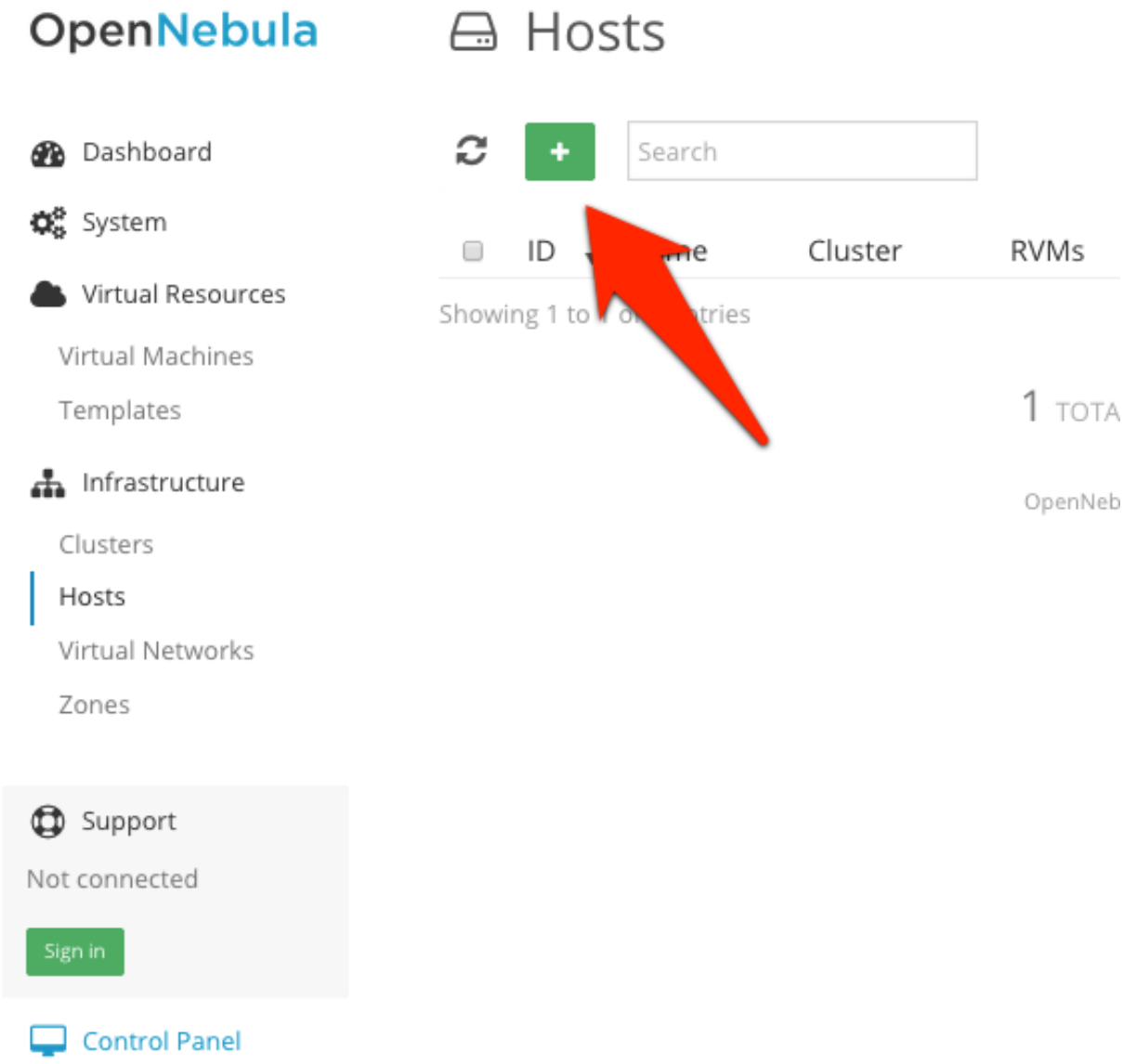
You will need the IP or hostname of the vCenter server, as well as an administrator credentials to successfuly import resources from vCenter.

3.3.1 Step 1. Sunstone login

Log in into Sunstone as **vOneCloud**, as explained in *the previous section*.

3.3.2 Step 2. Acquire vCenter Resources

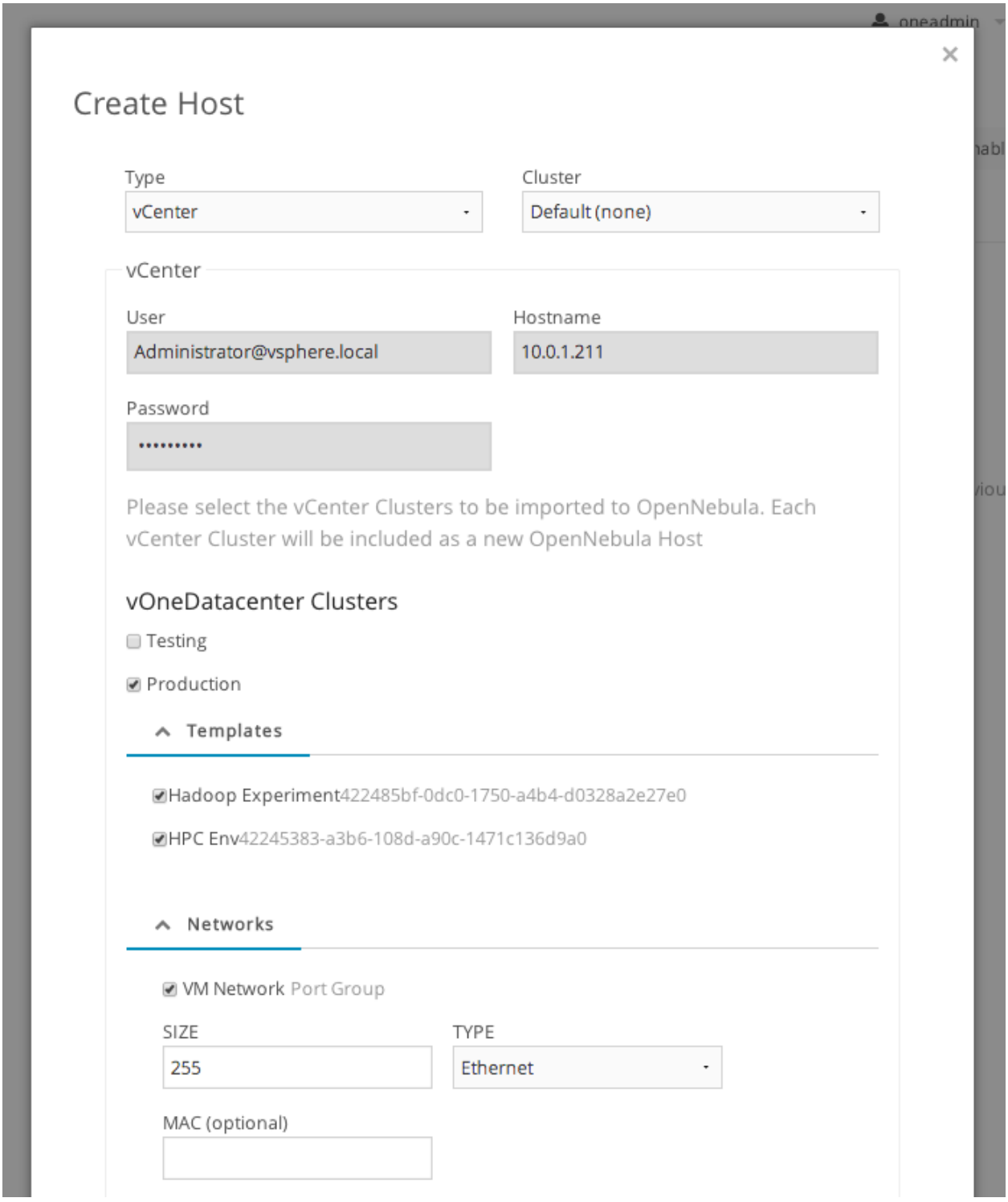
In Sunstone, proceed to the `Infrastructure --> Hosts` tab and click on the "+" green icon.



Warning: vOneCloud does not currently support spaces in vCenter cluster names

In the dialog that pops up, select vCenter as Type in the dropdown. You now need to fill in the data according to the following table:

Hostname	vCenter hostname (FQDN) or IP address
User	Username of a vCenter user with administrator rights
Password	Password for the above user



After the vCenter cluster is selected in Step 2, a paginated list of vCenter VM Templates and both Networks and Distributed vSwitches will be presented to be imported into vOneCloud. Select all the Templates, Networks and Distributed vSwitches you want to import, and vOneCloud will generate vOneCloud VM Template and Virtual Networks

resources representing the vCenter VM templates and vCenter Networks and Distributed vSwitches respectively. Networks, Distributes vSwitches and VM Templates resources imported from vCenter will have their names appended with a the name of the cluster where this resources belong in vCenter, to ease their identification within vOneCloud.

These vOneCloud VM templates can be modified selecting the VM Template in `Virtual Resources --> Templates` and clicking on the Update button, so the resulting VMs are adjusted to user needs. Among other options available through the Sunstone web interface:

- Information can be passed into the instantiated VM, through either *Contextualization or Customization*.
- Network interface cards can be added to give VMs access to different networks
- Capacity (MEMORY and CPU) can be modified
- VNC capabilities can be enabled

Note: VMs instantiated through vOneCloud will be named in vCenter as 'one-<vid>-<VM Name>', where <vid> is the id of the VM and VM Name is the name given to the VM in vOneCloud.

Also, Virtual Networks can be further refined with the inclusion of different Address Ranges. This refinement can be done at import time, defining the size of the network one of the following supported Address Ranges:

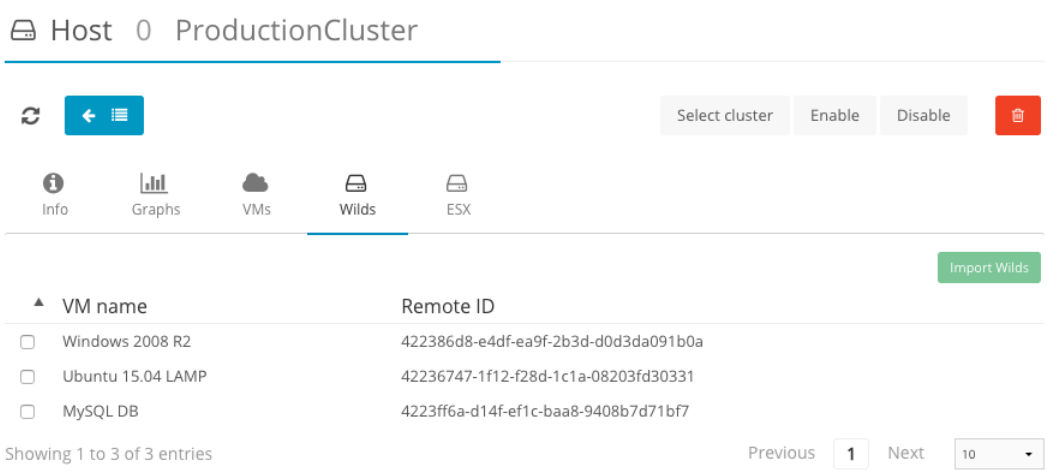
- IPv4: Need to define at least starting IP address. MAC address can be defined as well
- IPv6: Can optionally define starting MAC adddress, GLOBAL PREFIX and ULA PREFIX
- Ethernet: Does not manage IP addresses but rather MAC addresses. If a starting MAC is not provided, vOneCloud will generate one.

The networking information will also be passed onto the VM in the *Contextualization* process. Regarding the vCenter VM Templates and Networks, is important to take into account:

- vCenter **VM Templates with already defined NICs** that reference Networks in vCenter will be imported without this information in vOneCloud. These NICs will be invisible for vOneCloud, and therefore cannot be detached from the Virtual Machines. The imported Templates in vOneCloud can be updated to add NICs from Virtual Networks imported from vCenter (being Networks or Distributed vSwitches).
- We recommend therefore to use **VM Templates in vCenter without defined NICs**, to add them later on in the vOneCloud VM Templates

3.3.3 (Optional) Step 3. Import / Reacquire Virtual Machines, VM Templates and Networks

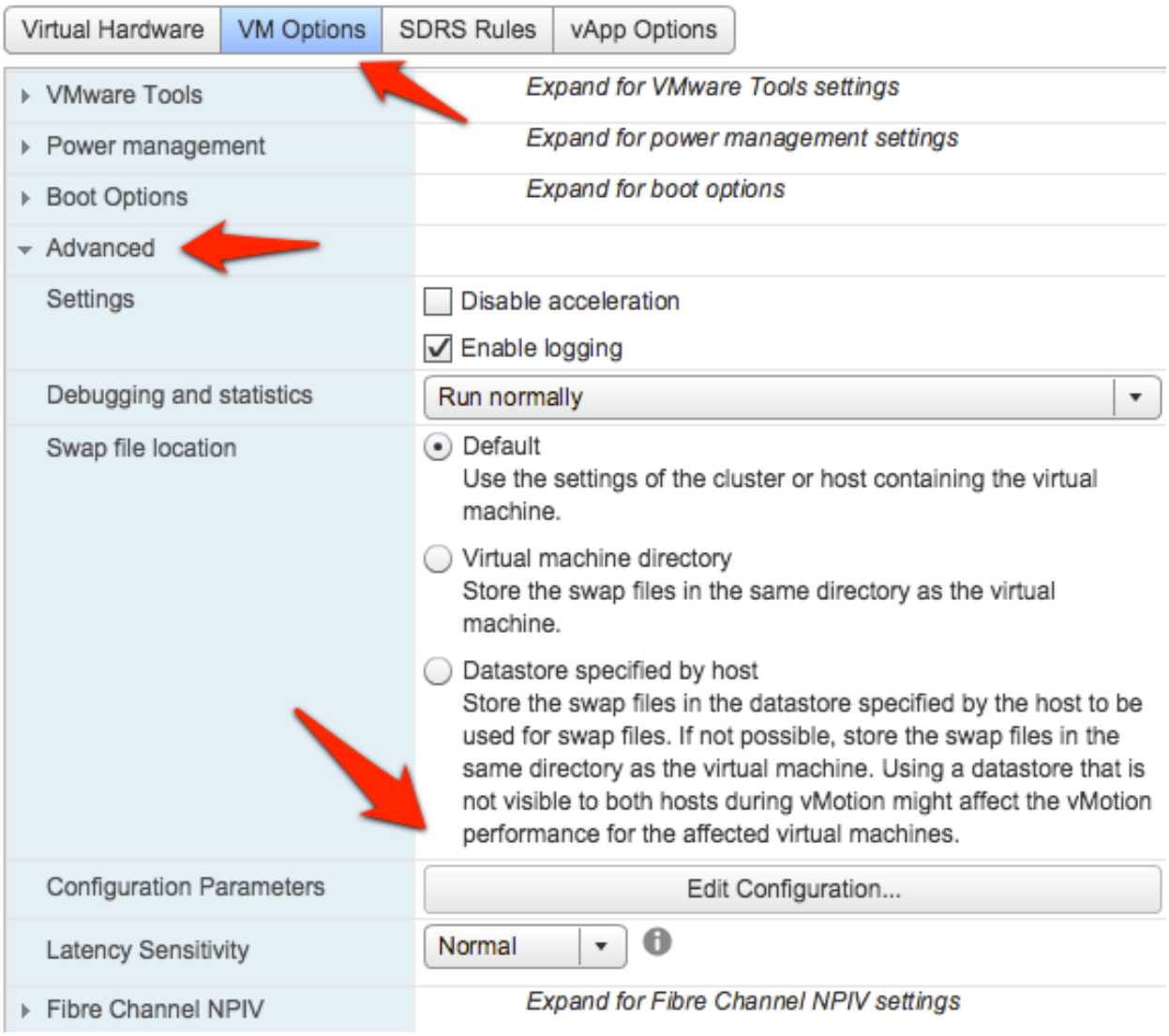
If the vCenter infrastructure has running (or powered off) Virtual Machines, vOneCloud can import and subsequently manage them. To import vCenter VMs, proceed to the WILDS tab in the Host info tab representing the vCenter cluster where the VMs are running in, select the VMs to be imported and click on the import button.



After the VMs are in the Running state, you can operate on their lifecycle, assign them to particular users, attach or detach network interfaces, create snapshots, do capacity resizing (change CPU and MEMORY after powering the VMs off), etc. All the funcionality that vOneCloud supports for regular VMs is present for imported VMs.

Note: This ability to import VMs also applies to hybrid hosts, it is possible through this same machanism to import VMs from SoftLayer, Azure and EC2.

Running VMs with open VNC ports are imported with the ability to stablish VNC connection to them via vOneCloud. To activate the VNC ports, you need to right click on the VM while it is shut down and click on “Edit Settings”, and set the `remotedisplay.*` settings show in the following images.

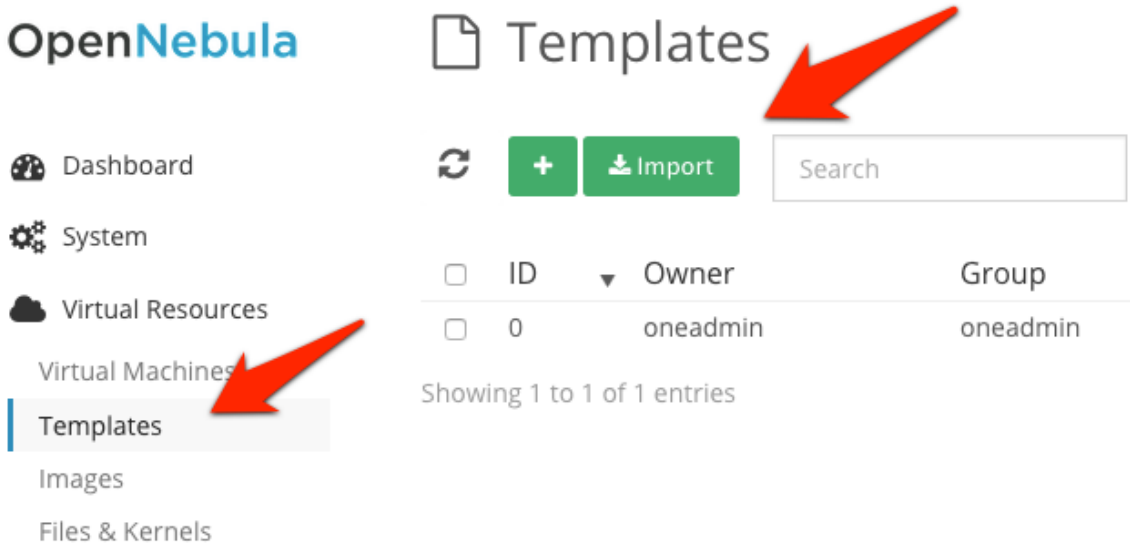


The following operations *cannot* be performed on an imported VM:

- Delete –recreate
- Undeploy (and Undeploy –hard)
- Migrate (and Migrate –live)
- Stop

vCenter VM Templates can be imported and reacquired using the `Import` button in `Virtual Resources -->`

`Templates`. Fill in the credentials and the IP or hostname of vCenter and click on the "Get Templates" button. Similarly, Networks and Distributed vSwitches can also be imported / reacquired from using a similar `Import` button in `Infrastructure --> Virtual Networks`.

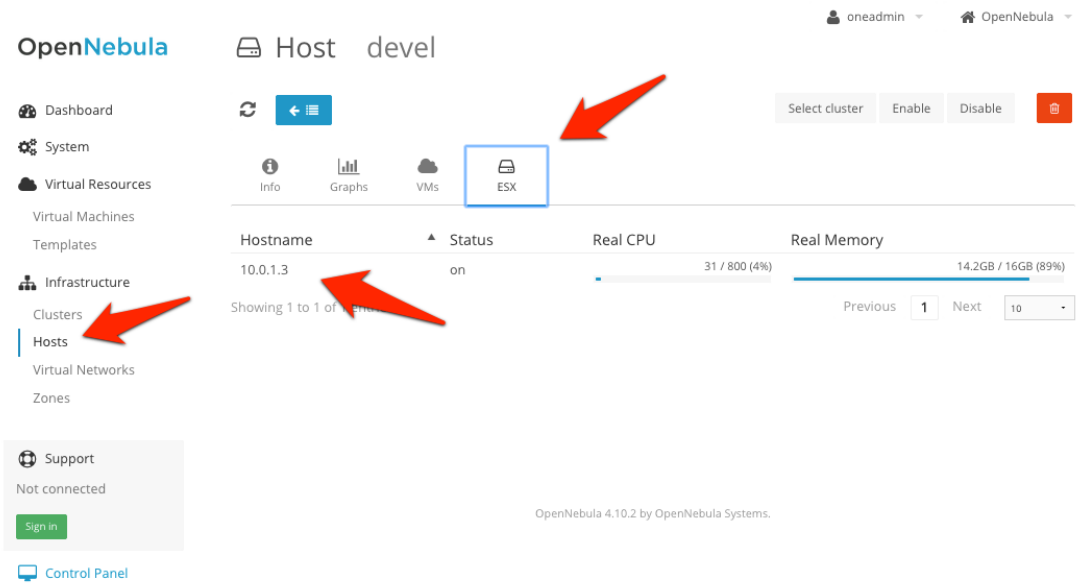


Note: The vCenter VM Templates, Networks, Distributed vSwitches and running Virtual Machines can be imported regardless of their position inside VM Folders, since vOneCloud will search recursively for them.

3.3.4 Step 4. Check Resources

Now it's time to check that the vCenter import has been succesful. In `Infrastructure --> Hosts` check vCenter has been imported, and if all the ESX hosts are available:

Note: Take into account that one vCenter cluster (with all its ESX hosts) will be represented as one vOneCloud host.



3.3.5 Step 5. Instantiate a VM Template

Everything is ready! Now vOneCloud is prepared to manage Virtual Machines. In Sunstone, go to `Virtual Resources --> Templates`, select one of the templates imported in **Step 2** and click on Instantiate. Now you will be able to control the lifecycle of the VM.

More information on available operations over VMs here.

3.4 Create a Virtual Datacenter

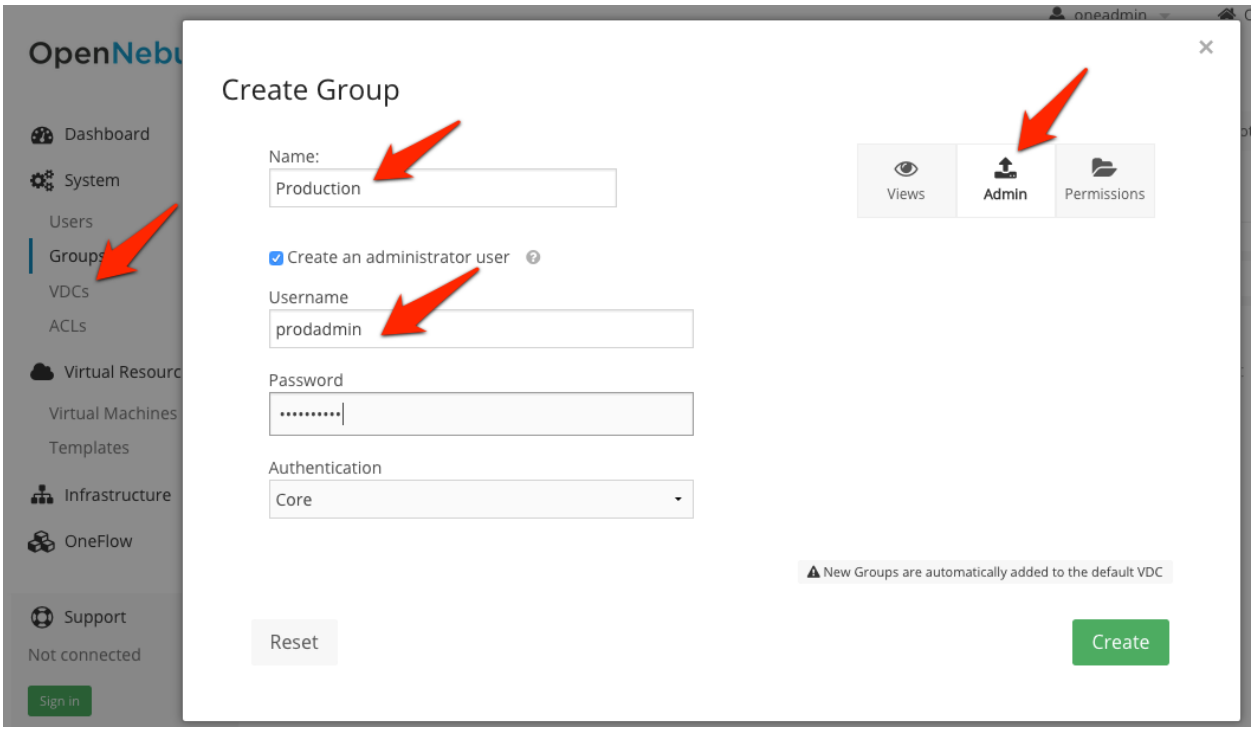
The provisioning model by default in vOneCloud is based on three different *roles* using three different web *interfaces*.

vOneCloud user comes preconfigured and is the **Cloud Administrator**, in full control of all the physical and virtual resources and using the vCenter view.

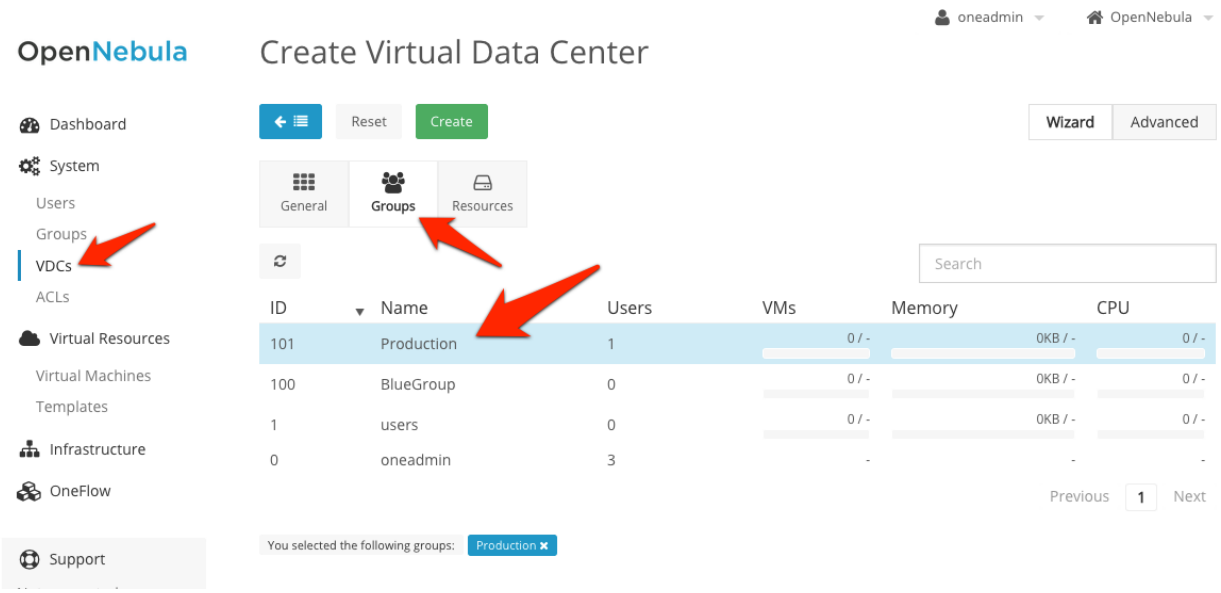
A Virtual Datacenter (VDC) defines an assignment of one or several groups to a pool of physical resources. This pool of physical resources consists of resources from one or several clusters, which are logical agroupations of hosts and virtual networks. VDCs are a great way to partition your cloud into smaller clouds, and asign them to groups with their administrators and users, completely isolated from other groups.

A **Group Admin** manages her partition of the cloud, including user management, but only within the VDCs assigned to the Group, not for the whole cloud like the **Cloud Administrator**.

Let's create a Group (under System) named *Production* with an administrator called **prodadmin**:

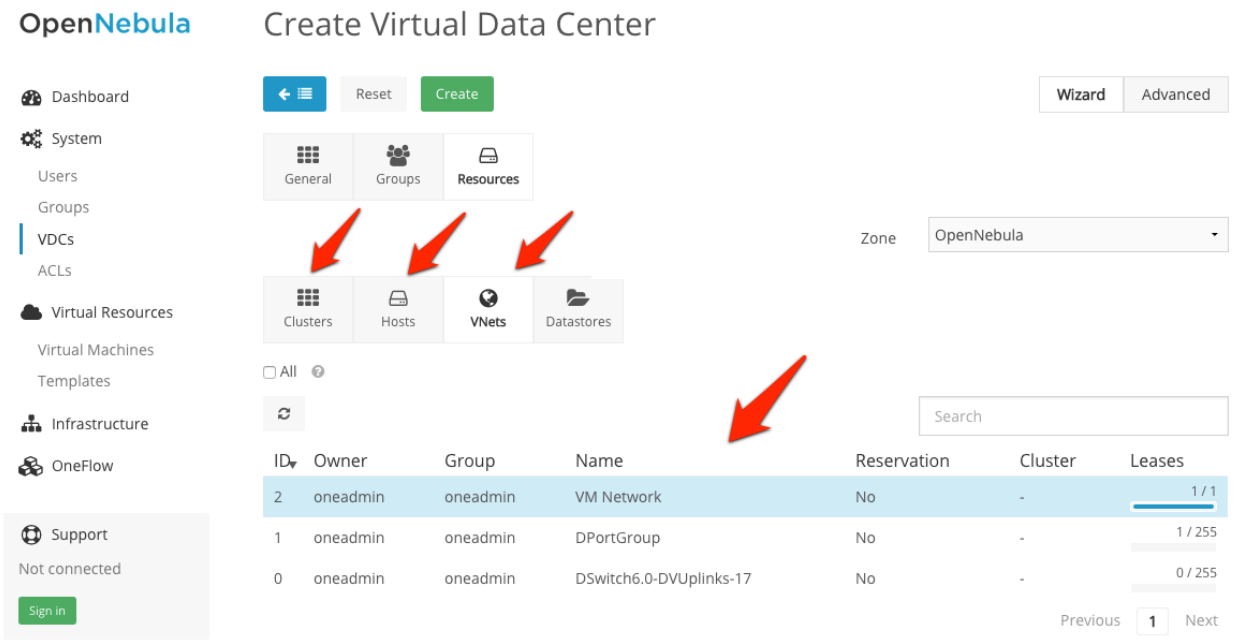


Let's create a VDCs (under System) named *ProductionVDC*, and assign the *Production* group to use it:

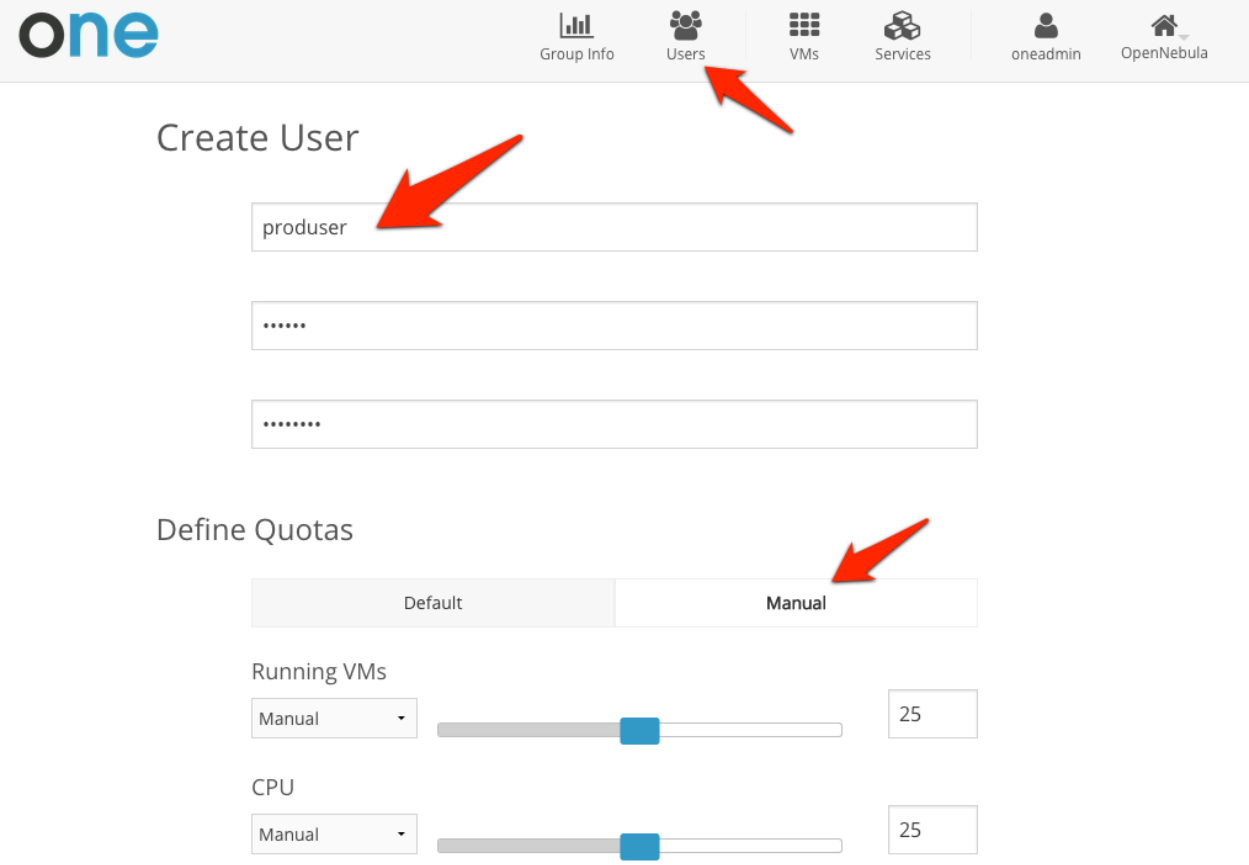


Let's add resources to the VDC under the "Resources" tab, for instance a vCenter instance and a Virtual Network:

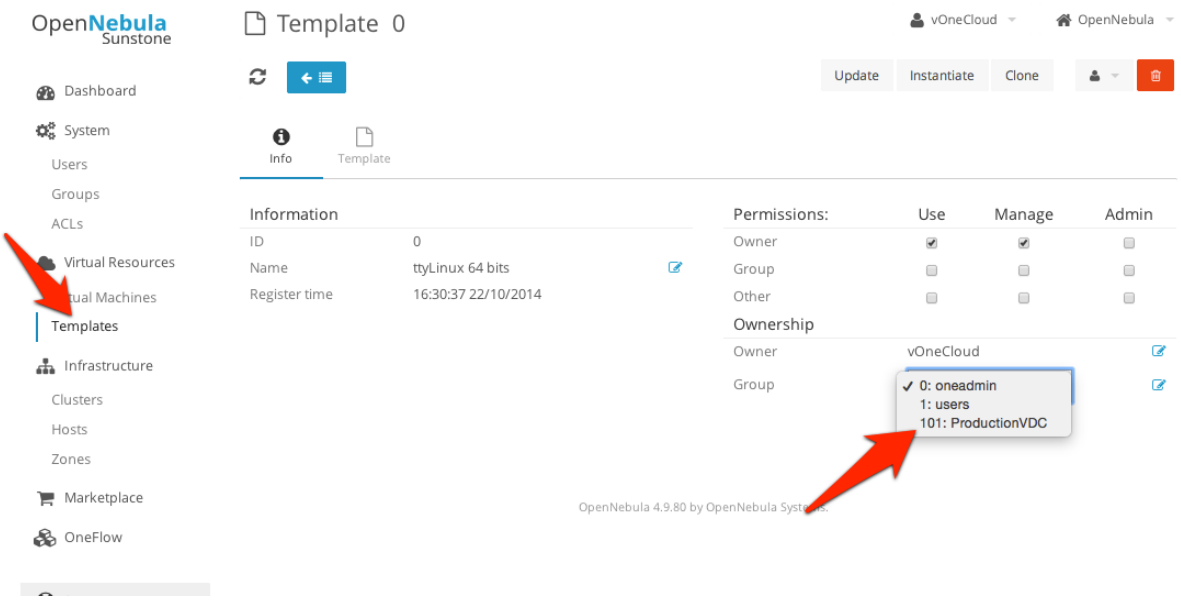
Note: Only Clusters, vCenters and Virtual Networks can be assigned to VDCs, current versions of vOneCloud do not manage vCenter datastores.



Now login again using the newly created **prodadmin**. The Group Admin view will kick in. Try it out creating the first *produser* and assign them quotas on resource usage:

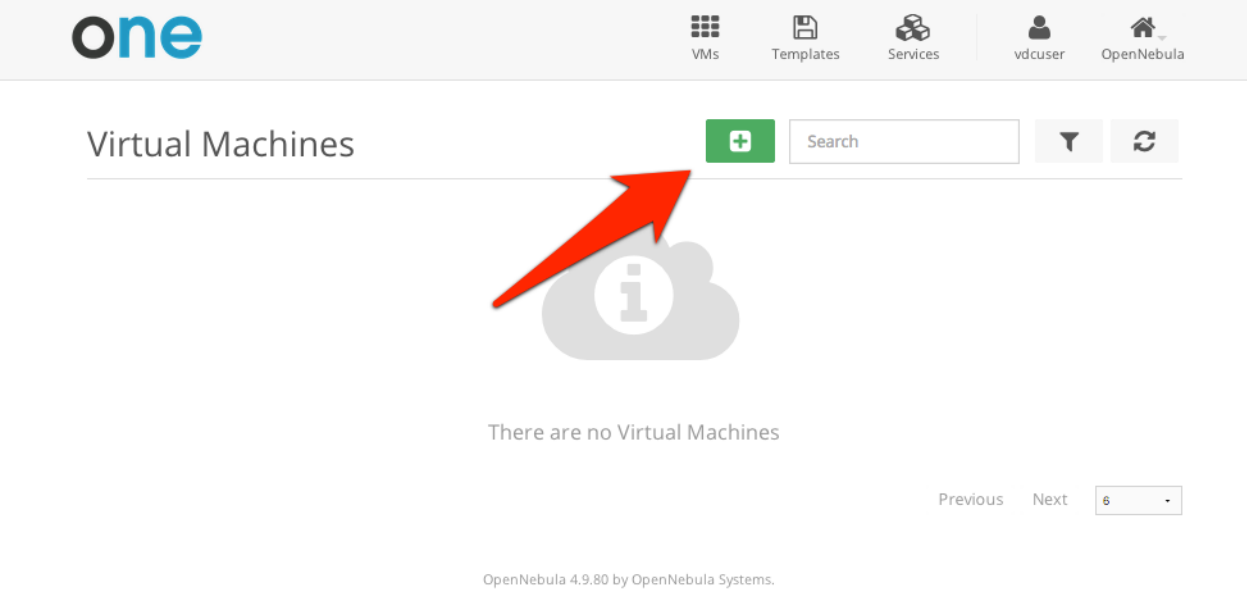


As *vOneCloud* user, in the vCenter View, you will be able to see all the VM Templates that have been automatically created when importing the vCenter infrastructure. You can assign any of these VM Templates to the VDC:



The same applies for Virtual Networks these VM Templates may use.

If you log with *produser*, the view will change to the vCenter Cloud View, where **vdcuser** can start consuming VMs based on the VM Template shared by the **Cloud Administrator** and allowed by the **vdcadmin**:



Read more about Group and VDC managing.

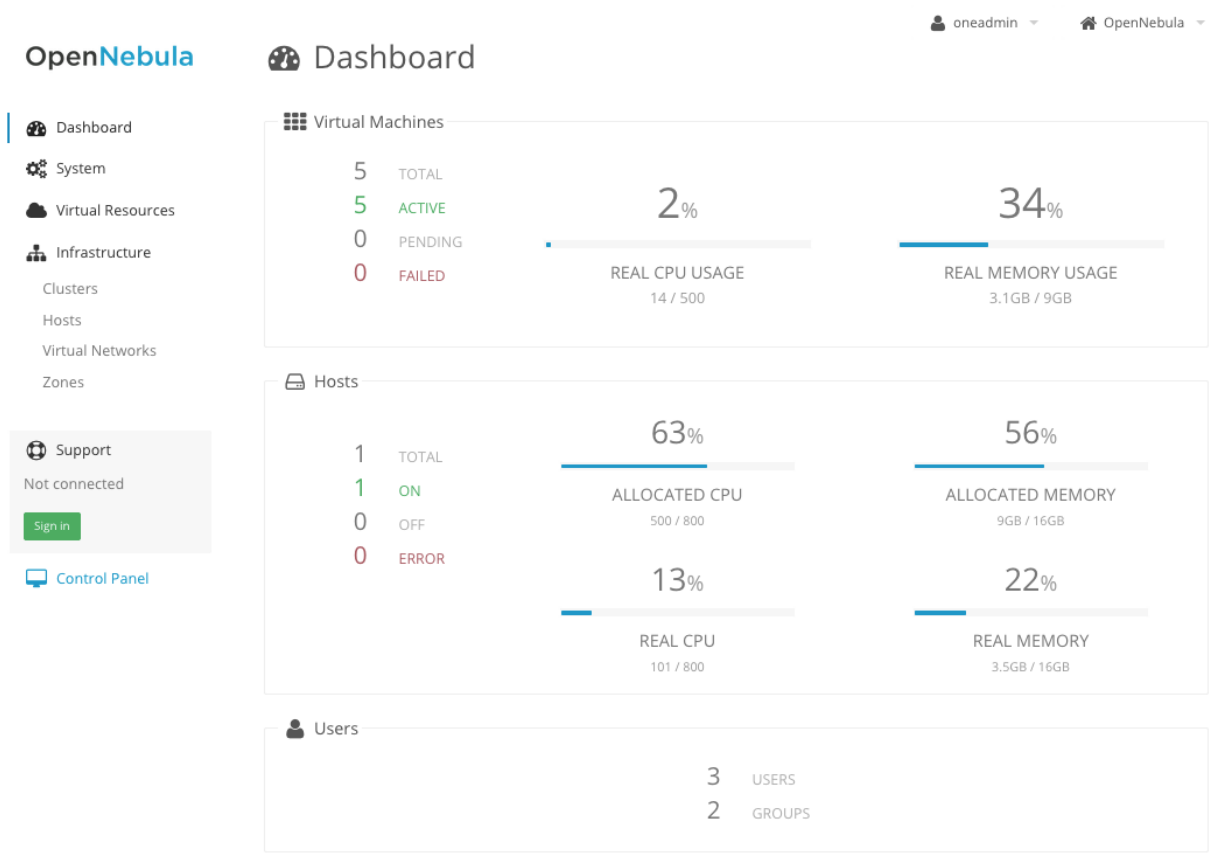
3.5 vOneCloud Interfaces

vOneCloud offers a rich set of interfaces to interact with your cloud infrastructure, tailored for specific needs of cloud administrators and cloud users alike.

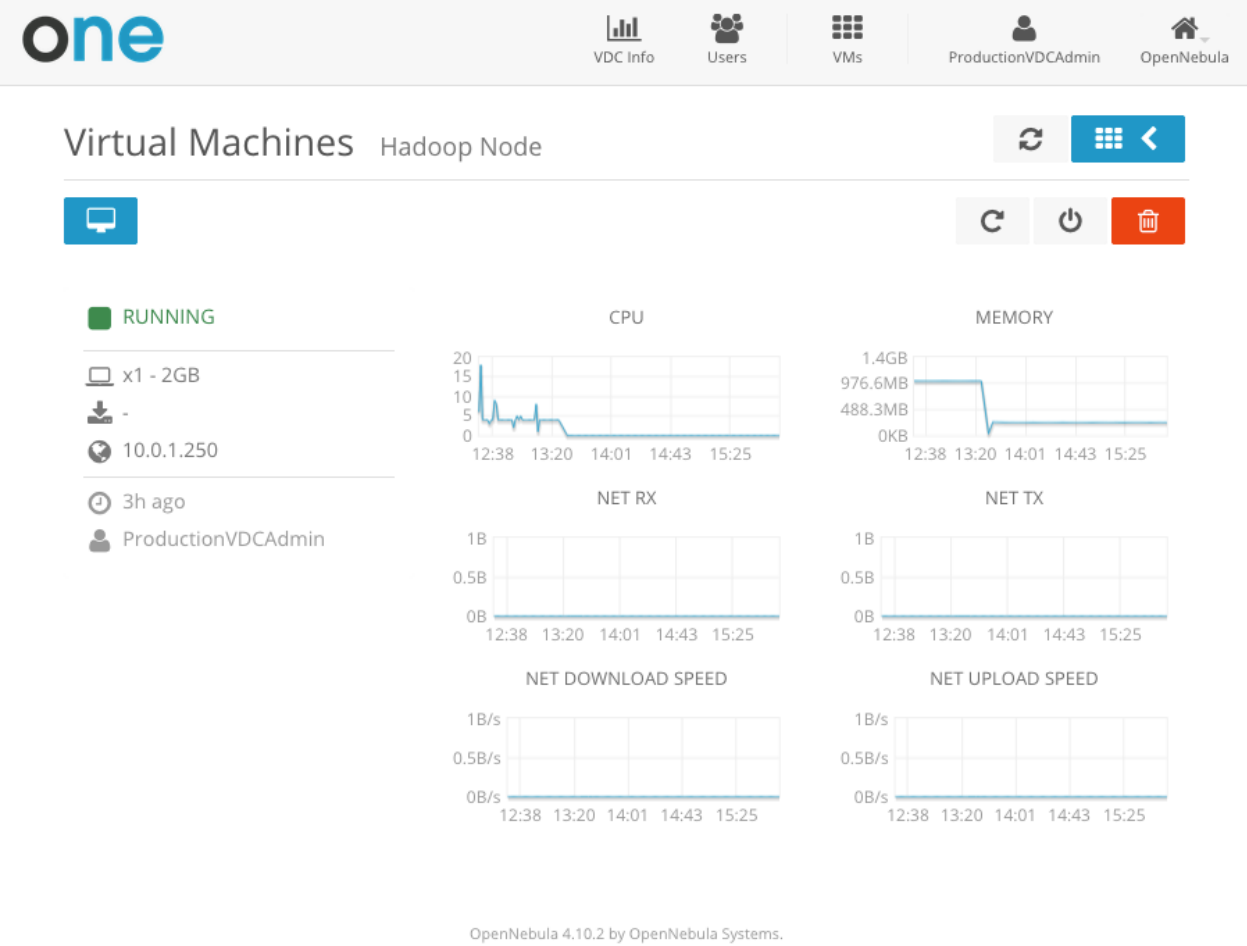
3.5.1 Web Interface (Sunstone)

vOneCloud web interface, called Sunstone, offers three main views:

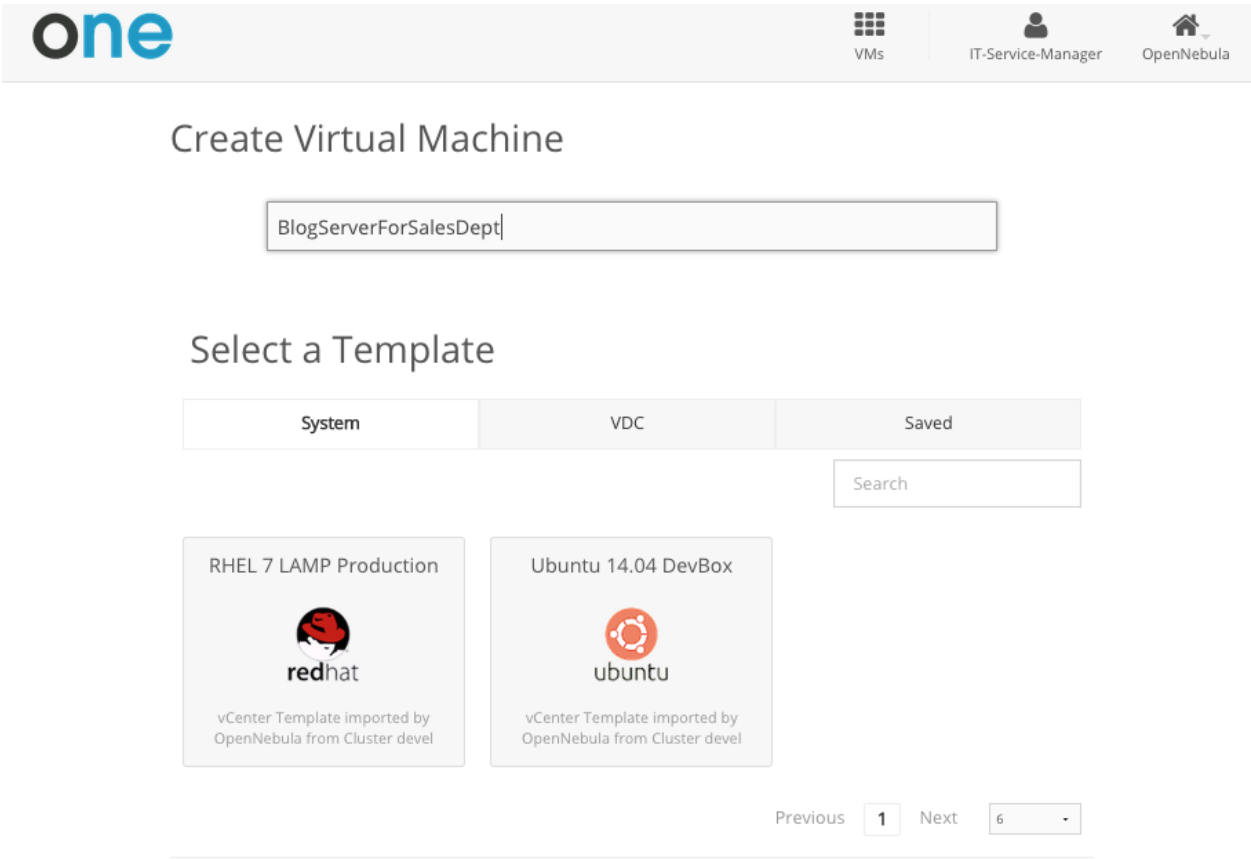
- Sunstone vCenter view: Aimed at cloud administrators, this view is tailored to present all the available options to manage the physical and virtual aspects of your vCenter infrastructure.



- Sunstone Group Admin View: Aimed at Group administrators, this interface is designed to manage all the virtual resources accesible by a group of users, including the creation of new users.



- **Sunstone vCenter Cloud View**: Aimed at end users, this interface eases virtual resource provisioning and hides all the complexity of the cloud that is going on behind the scenes. It is a tailored version of the Sunstone Cloud View, with adjusted functionality relevant to vOneCloud and vCenter.



3.5.2 Command Line Interface (CLI)

If you are a SysAdmin, you will probably appreciate vOneCloud's CLI, which uses the same design philosophy behind *nix commands (one command for each task).

Moreover, vOneCloud ships with a powerful tool (`onevcenter`) to import vCenter clusters, VM Templates and Networks. The tools is self-explanatory, just set the credentials and IP to access the vCenter host and follow on screen instructions.

To access the vOneCloud command line interface you need to *login into the vOneCloud appliance*, and switch to the *oneadmin* user.

3.5.3 Application Programming Interfaces (API)

If you are a DevOp, you are probably used to build scripts to automate tasks for you. vOneCloud offers a rich set of APIs to build scripts to perform these tasks in different programming languages:

- xmlrpc API Talk directly to the OpenNebula core
- Ruby OpenNebula Cloud API (OCA) Build tasks in Ruby
- Java OpenNebula Cloud API (OCA) Build tasks in Java

CHAPTER

FOUR

SECURITY AND RESOURCE CONSUMPTION CONTROL

4.1 Introduction

vOneCloud ships with several authentication plugins that can be configured to pull user data from existing authentication backends.

vOneCloud also implements a powerful permissions, quotas and ACLs mechanisms to control which users and groups are allowed to use which physical and virtual resources, keeping a record of the comsumption of these resources as well as monitoring their state periodically.

Take control of your cloud infrastructure!

4.2 Users, Groups and ACLs

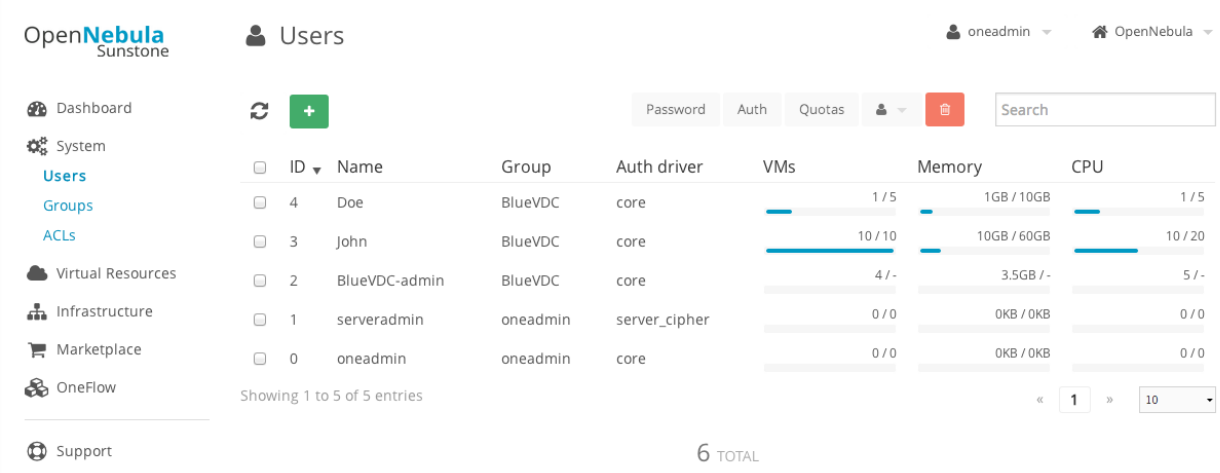
vOneCloud offers a powerful mechanism for managing, grouping and assigning roles to users. Permissions and Access Control List mechanisms ensures the ability to allow or forbid access to any resource controlled by vOneCloud, being physical or virtual.

4.2.1 User & Roles

vOneCloud can manage different types of users, attending to the permissions they have over infrastructure and logical resources.

User Type	Permissions	View
Cloud Administrators	enough privileges to perform any operation on any object	vcenter
Group Administrators	manage a limited set and users within VDCs	groupadmin
End Users	access a simplified view with limited actions to create new VMs	cloud

Note: VDC is the acronym for Virtual Datacenter

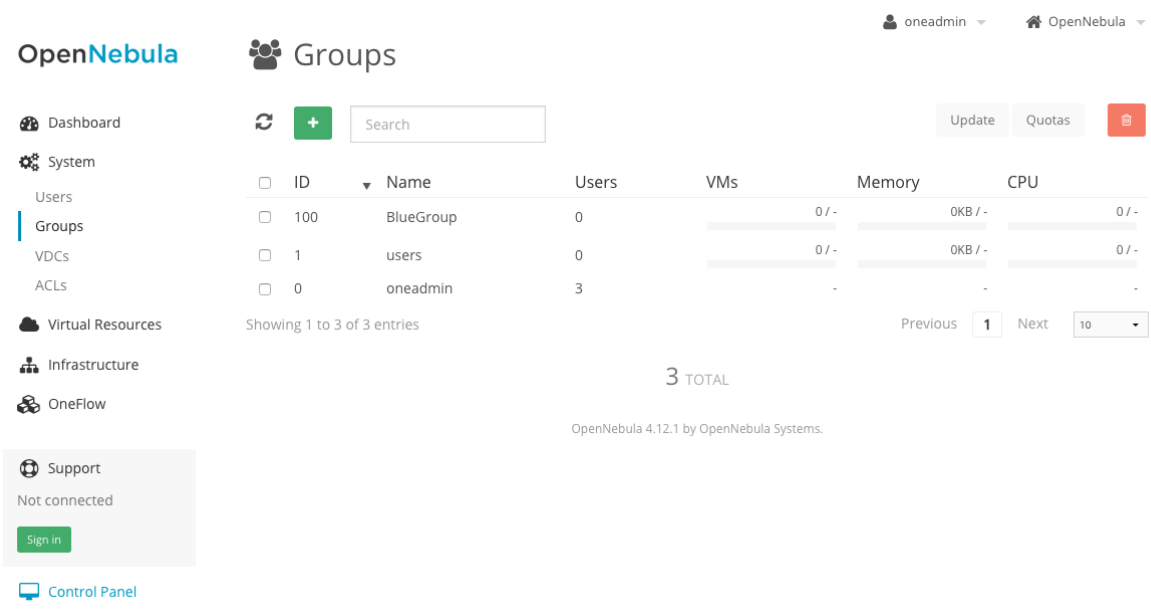


Learn more about user management here.

4.2.2 Group & VDC Management

A **group** of users makes it possible to isolate users and resources. A user can see and use the shared resources from other users. The **group** is an authorization boundary for the users, but you can also partition your cloud infrastructure and define what resources are available to each group using Virtual Data Centers (VDC).

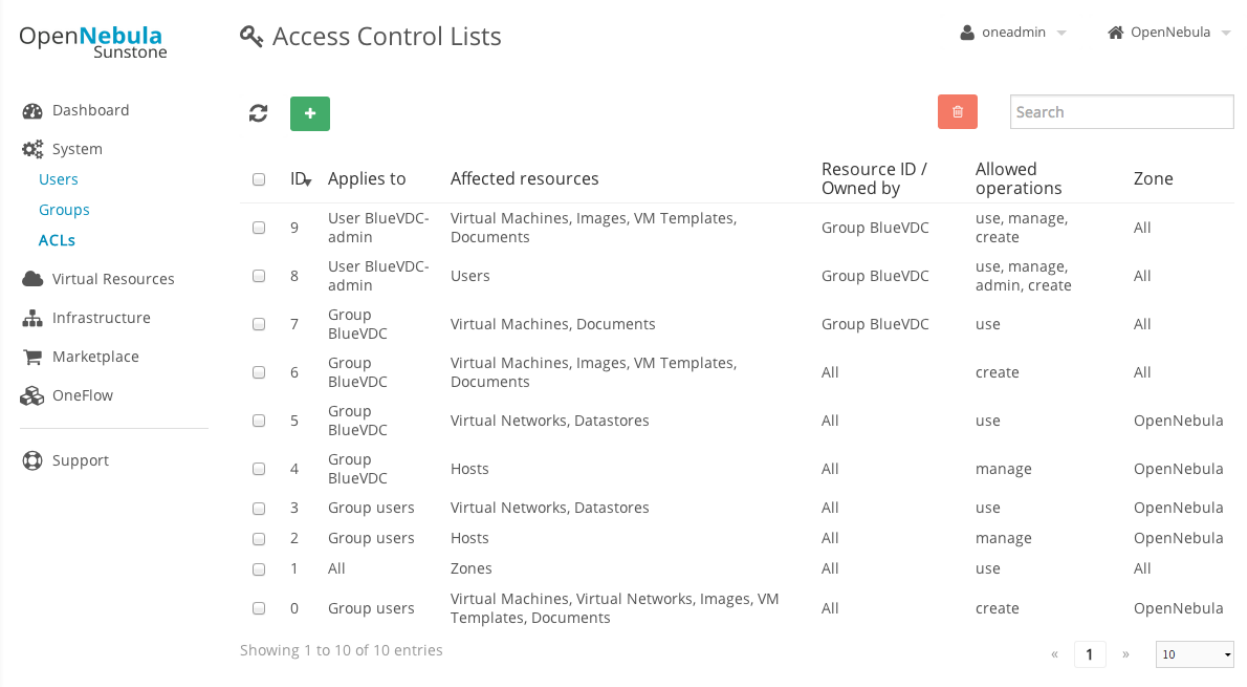
A VDC defines an assignment of one or several groups to a pool of physical resources. This pool of physical resources consists of resources from one or several clusters, which are logical agroupations of hosts and virtual networks. VDCs are a great way to partition your cloud into smaller clouds, and asign them to groups with their administrators and users, completely isolated from other groups.



Read more about groups and VDCs.

4.2.3 Access Control Lists

vOneCloud implements a very useful ACL mechanism that enables fine-tuning of allowed operations for any user, or group of users. Each operation generates an authorization request that is checked against the registered set of ACL rules. There are predefined ACLs that implements default behaviors (like VDC isolation), but they can be altered by the cloud administrator.



Writing (or even reading) ACL rules is not trivial, more information about ACLs here.

4.3 Resource Quotas

vOneCloud quota system tracks user and group usage of system resources, allowing the cloud administrator to set limits on the usage of these resources.

Quota limits can be set for:

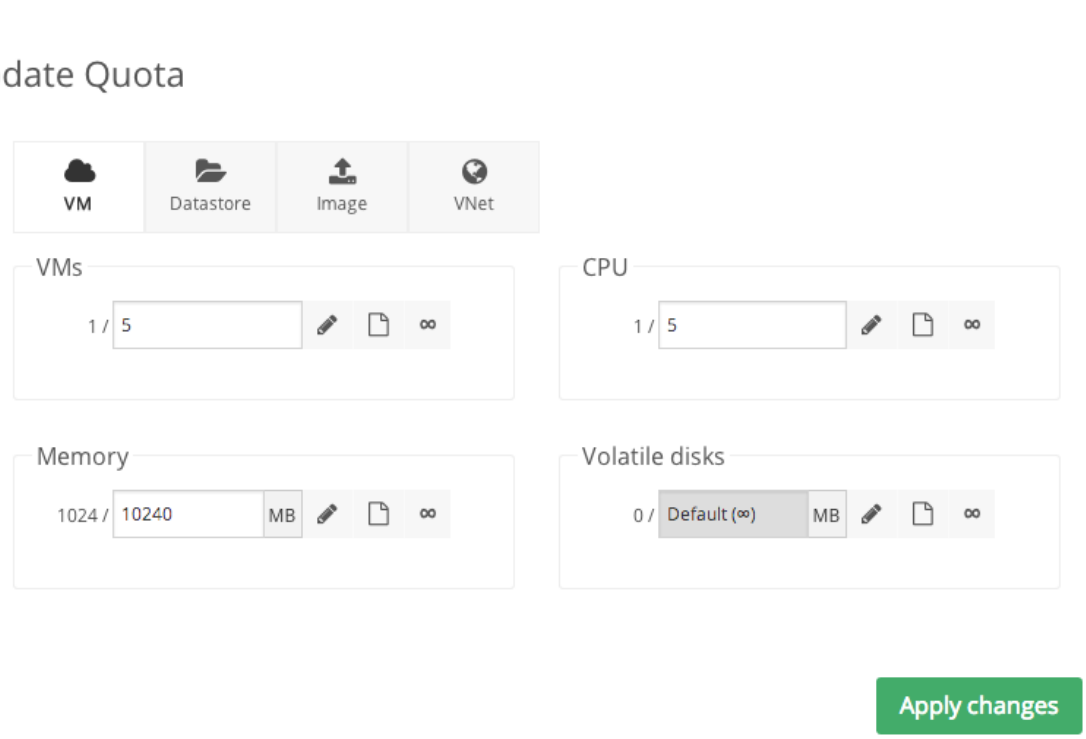
- **users**, to individually limit the usage made by a given user.
- **groups**, to limit the overall usage made by all the users in a given group.

Tracking the usage on:

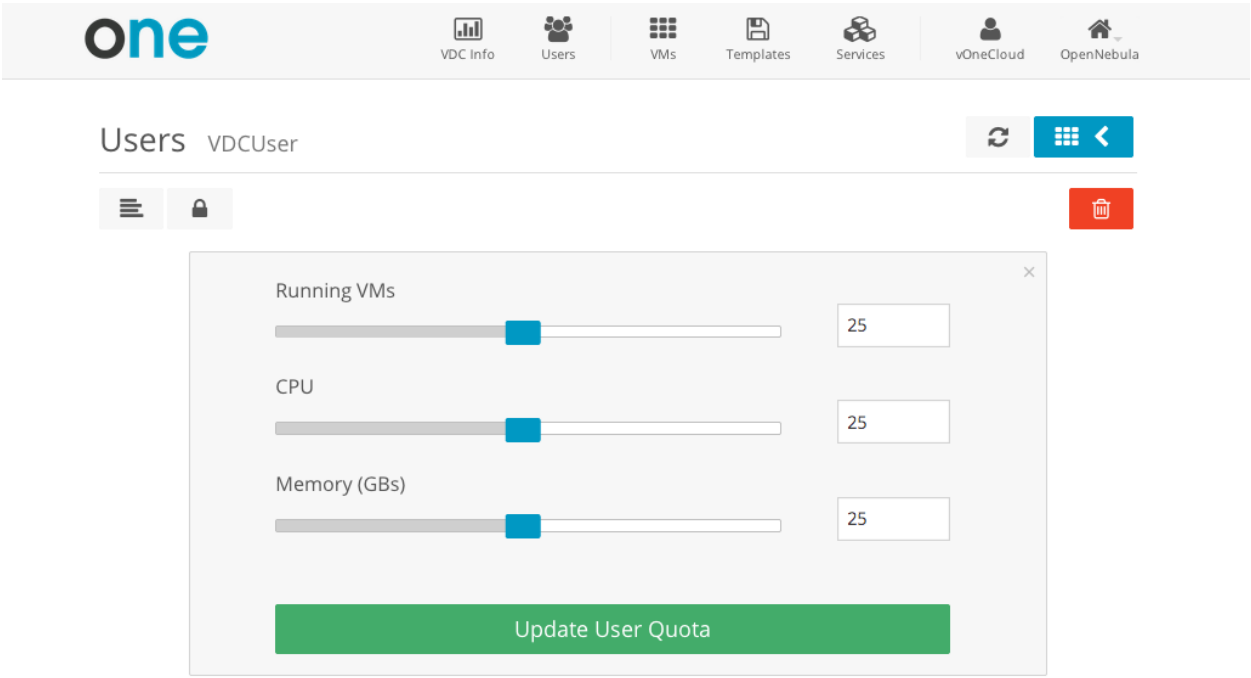
- **Compute**: Limit the overall memory, cpu or VM instances

> **Warning:** OpenNebula supports additional quotas for **Datastores** (control amount of storage capacity), **Network** (limit number of IPs), **Images** (limit VM instances per image). However these quotas are not available for the vCenter drivers.

Quotas can be updated either from the vCenter View:



Or from the Group Admin View:



Refer to this guide to find out more.

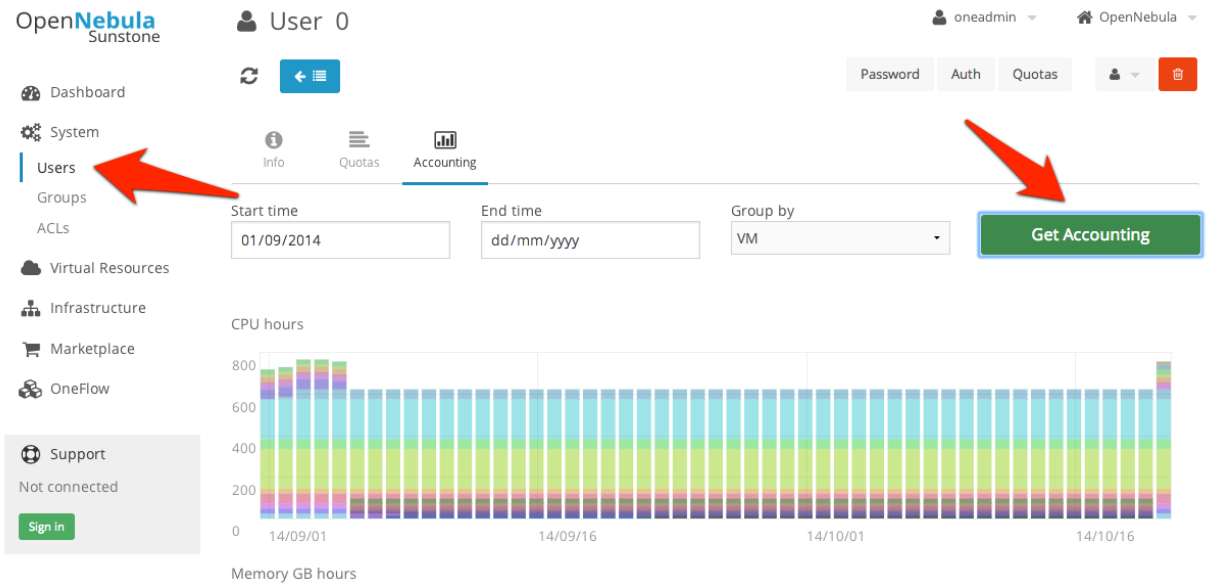
4.4 Accounting & Monitoring

vOneCloud is constantly monitoring the infrastructure resources to keep track of resource consumption. The objective is twofold: being able to have a clear picture of the infrastructure to aid in the resource scheduling, as well as being able to enforce *resource quotas* and give accounting information.

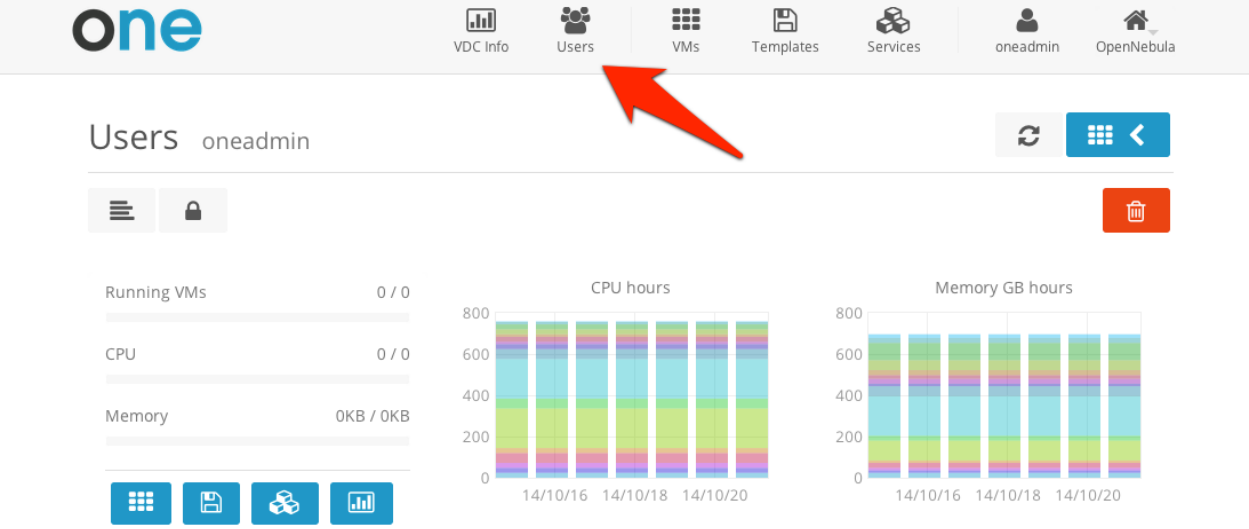
The monitoring subsystem gathers information relative to hosts and virtual machines, such as host and VM status, basic performance indicators and capacity consumption. vOneCloud comes preconfigured to retrieve such information directly from vCenter.

Using the information form the monitoring subsystem, vOneCloud is able to provide accounting information, both in text and graphically. An administrator can see the consumption of a particular user or group in terms of hours of CPU consumed, or total memory used in a given time window. This information is useful to feed a chargeback or billing platform.

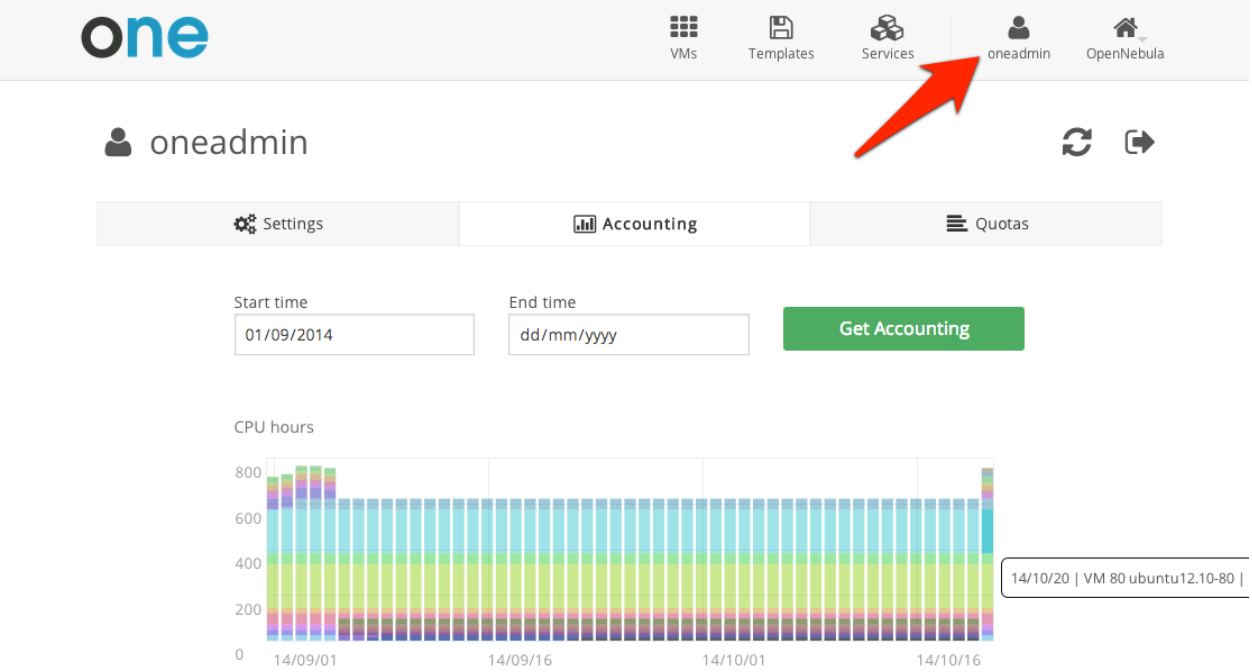
Accounting information is available from the vCenter View:



From the Group Admin View:



And from the vCenter Cloud View:



Learn more on the monitoring and accounting subsystems

4.5 Showback

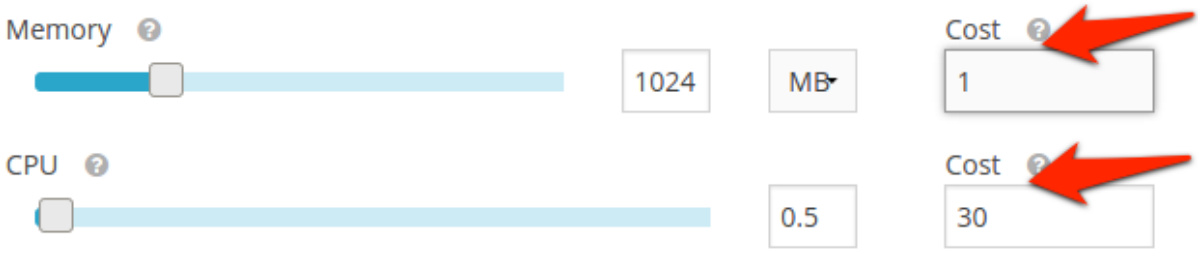
vOneCloud ships with functionality to report resource usage cost. Showback reports are genereted daily (at midnight)using the information retrieved from OpenNebula.

Set the VM Cost

Each VM Template can optionally define a cost. The cost is defined as cost per cpu per hour, and cost per memory

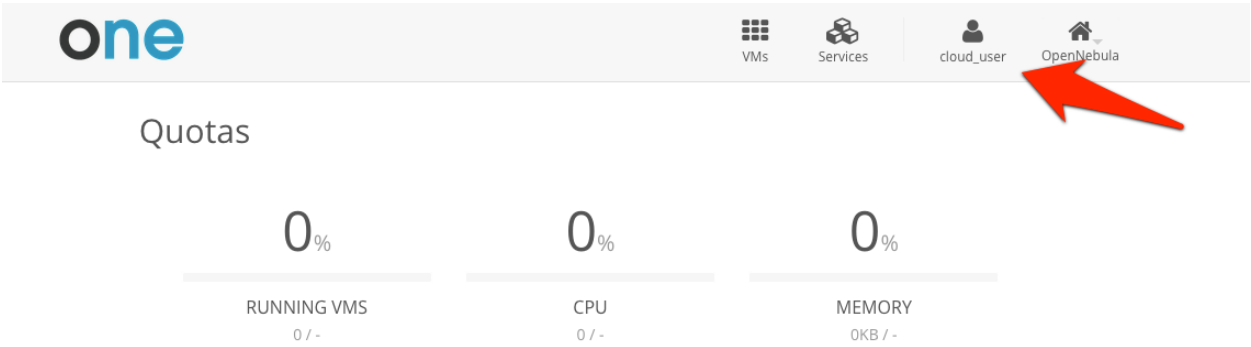
MB per hour. The cost units are abstract and their equivalent to monetary or other cost metrics have to be defined in each deployment.

This cost is defined per VM Template by the Cloud Administrator at the time of creating or updating a VM Template, applying a cost to the total Memory and CPU of the VMs that will be spawn from this VM Template.

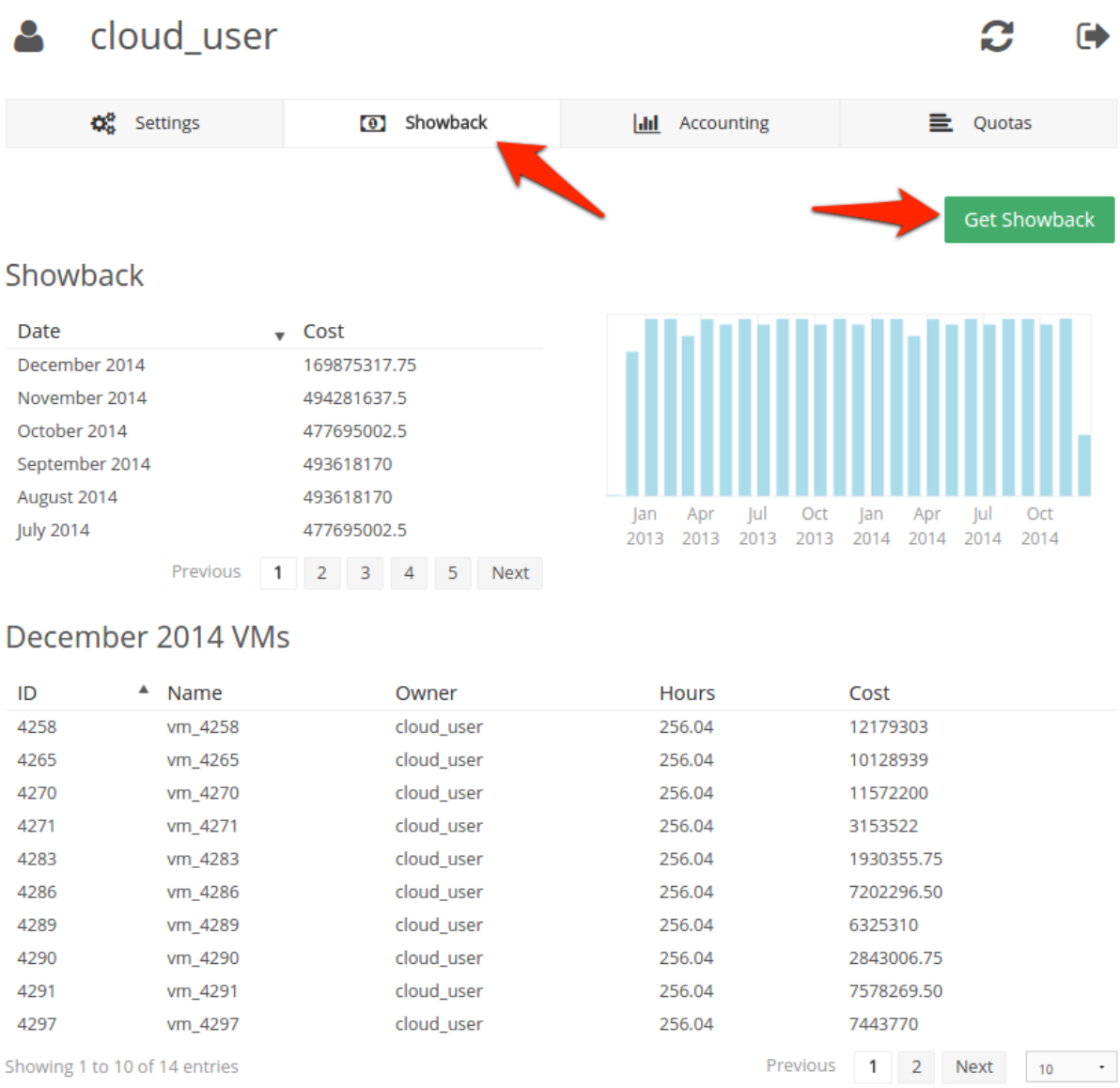


Retrieve Monthly Reports

Any user or administrator can see their monthly showback reports clicking on their user icon to access Settings.



And clicking on the Showback tab, obtain the cost consumed by clicking on the “Get Showback”



Learn more on the Showback functionality.

CHAPTER

FIVE

GUEST CONFIGURATION

5.1 Introduction

vOneCloud will use pre configured vCenter VM Templates, which leverages the functionality provided by vCenter to build such templates. Additionally, vOneCloud provides functionality to tailor the VM guest Operating System to adjust it for the end user needs. OpenNebula povides two mechanisms to configure the newly created VMs.

- *OpenNebula Contextualization*: It allows configuration and information sharing between the vOneCloud interface and the Virtual Machine
- *vCenter Customization Specifications*: It ties the template with a vCenter Customization Specification so it is configured on VM creation

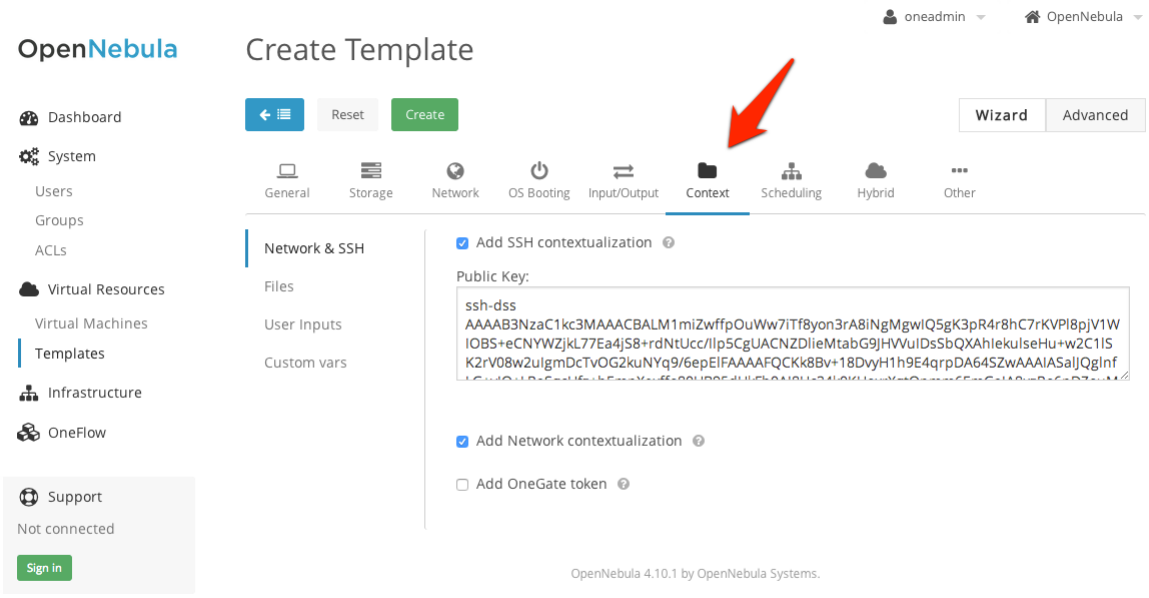
Warning: These options can not be used together. A template can use either OpenNebula Contextualization or vCenter customization.

This section will instruct on the needed actions to be taken into account to build vOneCloud Templates to deliver cloud users with personalized and perfectly adjusted Virtual Machines.

5.2 Building a Template for Contextualization

In order to pass information to the instantiated VM template, the Context section of the vOneCloudVM Template can be used. These templates can be updated in the Virtual Resources -> Templates tab of the vOneCloud GUI, and they can be updated regardless if they are *directly imported from vCenter* or *created through the vOneCloud Templates tab*.

Note: Installing the Contextualization packages in the Virtual Machine image is required to pass this information to the instantantiated VM template. Make sure you follow the *Guest Contextualization* guide to properly prepare your VM templates.



Warning: Passing files and network information to VMs through contextualization is currently not supported

Different kinds of context information can be passed onto the VMs:

5.2.1 Network & SSH

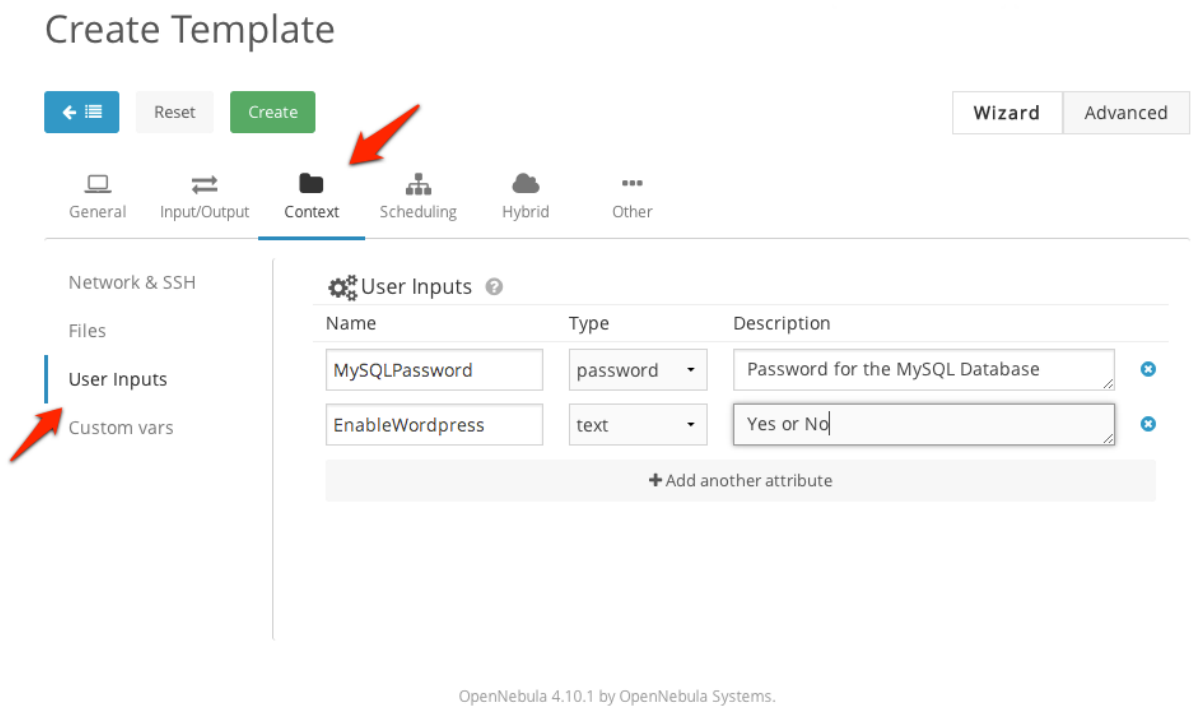
Networking information can be passed onto the VM, namely the information needed to correctly configure each one of the VM network interfaces.

You can add here an public keys that will be available in the VM at launch time to configure user access through SSH.

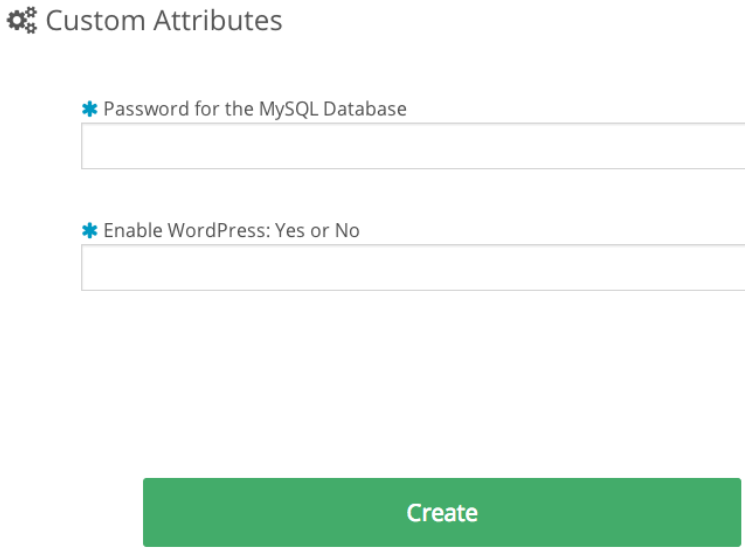
5.2.2 User Inputs

These inputs are a special kind of contextualization that built into the templates. At instantiation time, the end user will be asked to fill in information for the defined inputs, and the answers will be packed and passed onto the VM.

For instance, vOneCloud adminsitrator can build a VM Template that will ask for the MySQL password (the MySQL software will be configured at VM boot time and this password will be set) and for instance whether or not to enable WordPress:



The end user will then be presented with the following form when instantiating the previously defined VM Template



5.2.3 Custom vars

These are personalized information to pass directly to the VM, in the form of Key - Value.

There are two special custom variables which are very useful for VM software provisioning. They are called START_SCRIPTS and START_SCRIPTS_BASE64 (they are equivalent, only the latter gets base64 decoded before

being executed at boot time). They can be use for any general purpose (pull a Chef recipe and running it for instance, configure a software component, perform a configuration task in the OS, etc). For instance, the following will add a DNS name **nfs-share** pointing to a particular IP in a Linux based VM:

START_SCRIPT = "echo 'nfs-share 10.0.1.23' > /etc/hosts"

5.3 Guest Contextualization

The information defined at the *VM Template building* time is presented to the VM using the VMware VMCI channel. This information comes encoded in base64 can be gathered using the VMware Tools.

Note: VMware tools are needed in the guestOS to enable several features (contextualization and networking feedback). Please install VMware Tools (for Windows) or Open Virtual Machine Tools (for *nix) in the guestOS.

In order to make your VMs aware of OpenNebula, you **must** install the *official packages*. Packages for both Linux and Windows exist that can collect this data and configure the supported parameters.

Parameter	Description
SET_HOST	Change the hostname of the VM. In Windows the machine needs to be restarted.
SSH_PUBLIC_KEY	SSH public keys to add to authorized_keys file. This parameter only works with Linux guests.
USERNAME	Create a new administrator user with the given user name. Only for Windows guests.
PASSWORD	Password for the new administrator user. Used with `USERNAME` and only for Windows guests.
DNS	Add DNS entries to `resolv.conf` file. Only for Linux guests.
NETWORK	If set to "YES" vOneCloud will pass Networking for the different NICs onto the VM
START_SCRIPT	Shell script to be executed at boot time by the VM to provision software within it
START_SCRIPT_BASE64	Same as START_SCRIPT but will be base64 decoded prior to be executed

In Linux guests, the information can be consumed using the following command (and acted accordingly):

```
$ vmtoolsd --cmd 'info-get guestinfo.opennebula.context' | base64 -d
MYSQLPASSWORD = 'MyPassword'
ENABLEWORDPRESS = 'YES'
```

5.3.1 Linux Packages

The following Linux distributions are supported supported:

- CentOS/RedHat 6 and 7
- Debian 7 and 8
- Ubuntu from 12.04 to 15.05

The linux packages can be downloaded from its project page and installed in the guest OS. There is one rpm file for Debian and Ubuntu and an rpm for RHEL and CentOS. After installing the package shutdown the machine and create a new template.

5.3.2 Windows Package

The Windows versions are supported:

- Windows 2008 R2

- Windows 2012
- Windows 2012 R2

The official addon-opennebula-context provides all the necessary files to run the contextualization in Windows 2008 R2.

The contextualization procedure is as follows:

1. Download `startup.vbs` and `context.ps1` to the Windows VM and save them in `C:\`.
2. Open the Local Group Policy Dialog by running `gpedit.msc`. Under: Computer Configuration -> Windows Settings -> Scripts -> startup (right click); browse to the `startup.vbs` file and enable it as a startup script.

After that power off the VM and create a new template from it.

5.4 vCenter Customization

vCenter offers a way to prepare the guest OS on boot. For example configuring its network, licenses, Active Directory server, etc. OpenNebula vCenter drivers offers a way to tie one OpenNebula template with one of these customizations so it is applied on VM startup. You can get more information about this system in VMware documentation.

There are a couple of things to take into account:

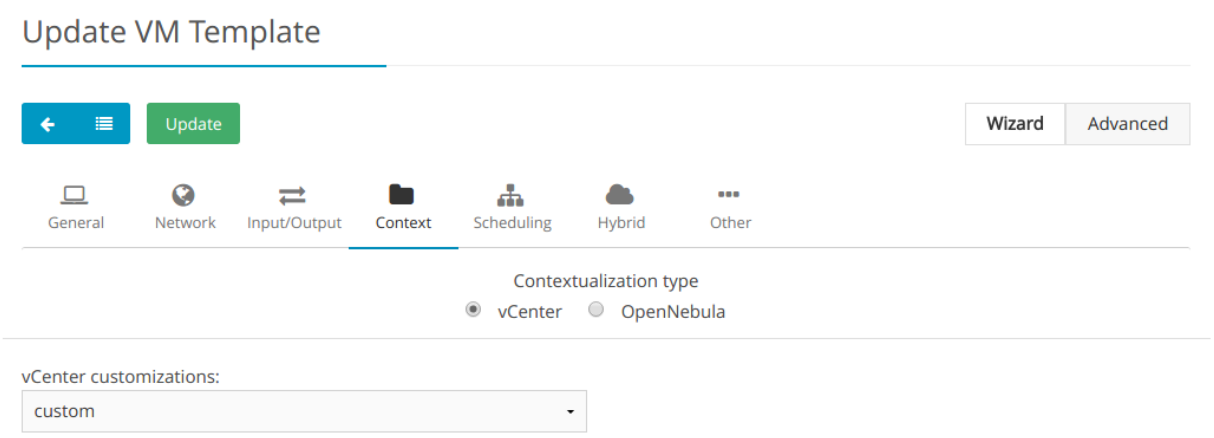
- This system is not compatible with OpenNebula contextualization as this customization overwrites the networking changes made by context scripts.
- VM network configuration must be done externaly to OpenNebula. Either with a DHCP server or manually setting IPs for each interface.
- This method can be used in all the Guest OSs supported by vCenter.

5.4.1 Template Customization Using Sunstone

For vcenter templates there are two options in the context tab. To use vCenter Customization select “vCenter” in the as “Contextualization type”. This will show a dropdown with all the customizations from all the hosts. There you can select from these possibilities:

- **None**: No customization will be applied
- **Custom**: You will be able to type manually the name of one customization
- The name of customizations found in vCenters

Make sure that the customization applied is available in the vCenter where the VM template reside.


Update VM Template
Update
Wizard
Advanced
General
Network
Input/Output
Context
Scheduling
Hybrid
Other
Contextualization type
vCenter
OpenNebula
vCenter customizations:
custom

CHAPTER

SIX

INFRASTRUCTURE CONFIGURATION

6.1 Introduction

Now that you are familiar with vOneCloud concepts and operations, it is time to extend its functionality by adding new infrastructure components and/or configuring options that do not come enabled by default in vOneCloud, but are present in the software nonetheless.

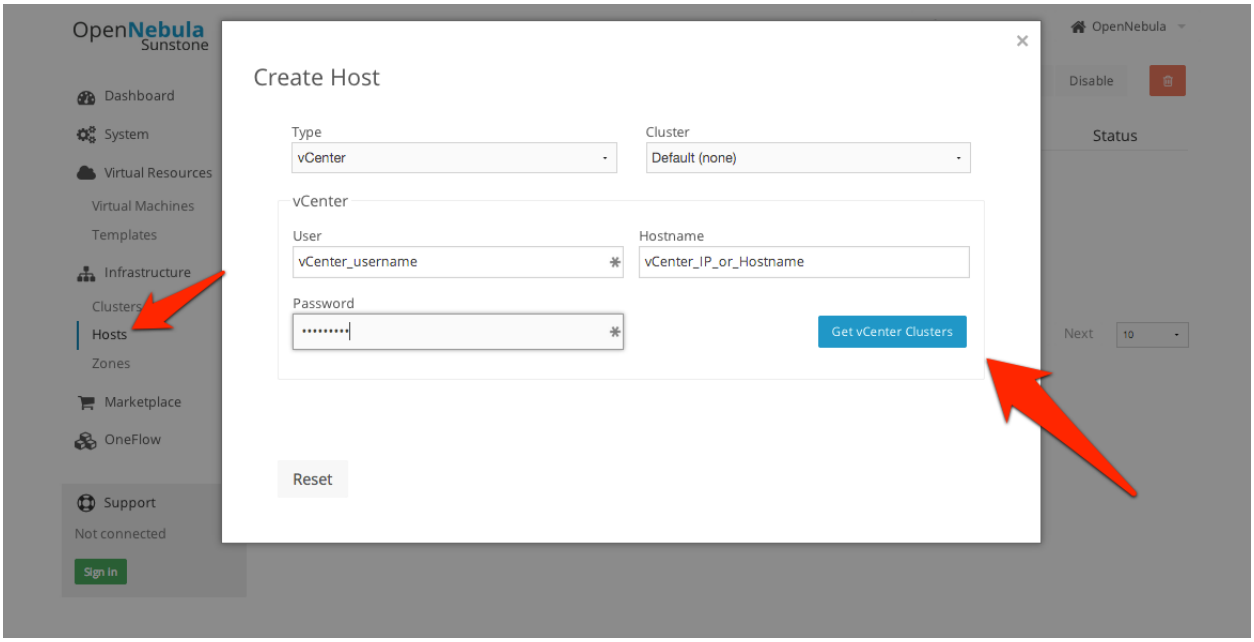
6.2 Add New vCenters, VM Templates and Networks

vOneCloud can manage an unlimited number of vCenters. Each vCenter is going to be represented by an vOneCloud host, which in turn abstracts all the ESX hosts managed by that particular instance of vCenter.

The suggested usage is to build vOneCloud templates for each VM Template in each vCenter. The built in scheduler in vOneCloud will decide which vCenter has the VM Template needed to launch the VM.

6.2.1 Add New vCenter Cluster

The mechanism to add a **new vCenter** is exactly the same as the one used to *import the first one into vOneCloud*. It can be performed graphically from the vCenter View:



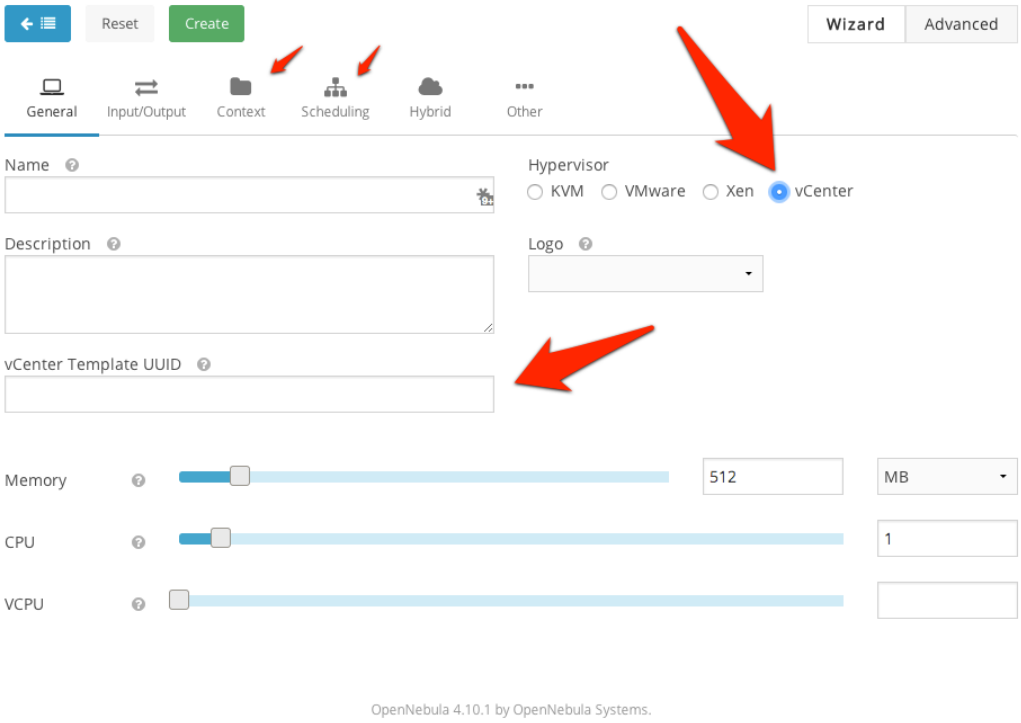
Note: vOneCloud will create a special key at boot time and save it in /var/lib/one/.one/one_key. This key will be used as a private key to encrypt and decrypt all the passwords for all the vCenters that vOneCloud can access. Thus, the password shown in the vOneCloud host representing the vCenter is the original password encrypted with this special key.

6.2.2 Add New VM Template

To create a new **vOneCloud VM Template**, let's see an example:

> Firsts things first, to avoid misunderstandings, there are two VM templates we will refer to: the vOneCloud VM Templates and the vCenter VM Templates. The formers are created in the vOneCloud web interface (Sunstone), whereas the latters are created directly through the vCenter Web Client.
>
> A cloud administrator builds two vOneCloud templates to represent one vCenter VM Template avaiable in vCenterA and another available in vCenterB. As previous work, the cloud administrator creates two vCenter VM templates, one in each vCenter.
>
> To create a vOneCloud VM Template representing a vCloud VM Template, log in into Sunstone as **vOneCloud** user as in explained *here*, proceed to the `Virtual Resources -> Templates`, and click on the **+** sign. Select *vCenter* as the hypervisor, and type in the *vCenter Template UUID*. You can also set a capacity (CPU and Memory) that would be honored at the time of instantiating the VM. In the *Scheduling* tab you can select the hostname of the specific vCenter. The *Context* tab allows to pass information onto the VM to tailor it for its final use (read more about it *here*). In *Network* tab a valid Virtual Network (see below) can added to the VM, possible values for the MODEL type of the network card are:
>
> - virtuale1000
> - virtuale1000e
> - virtualpcnet32
> - virtualsriovethernetcard
> - virtualvmxnetm

- virtualvmxnet2
- virtualvmxnet3



Fill in with UUID **uuidA** in and select host vCenterA. Repeat for vCenterB.

If a user instantiates one of these templates, the vOneCloud scheduler will pick the right vCenter in which to instantiate the VM Template.

The variable KEEP_DISKS_ON_DONE can be used in the VM template to instruct vOneCloud not to erase the VM disks after it enters the DONE state (either through shutdown or cancel)

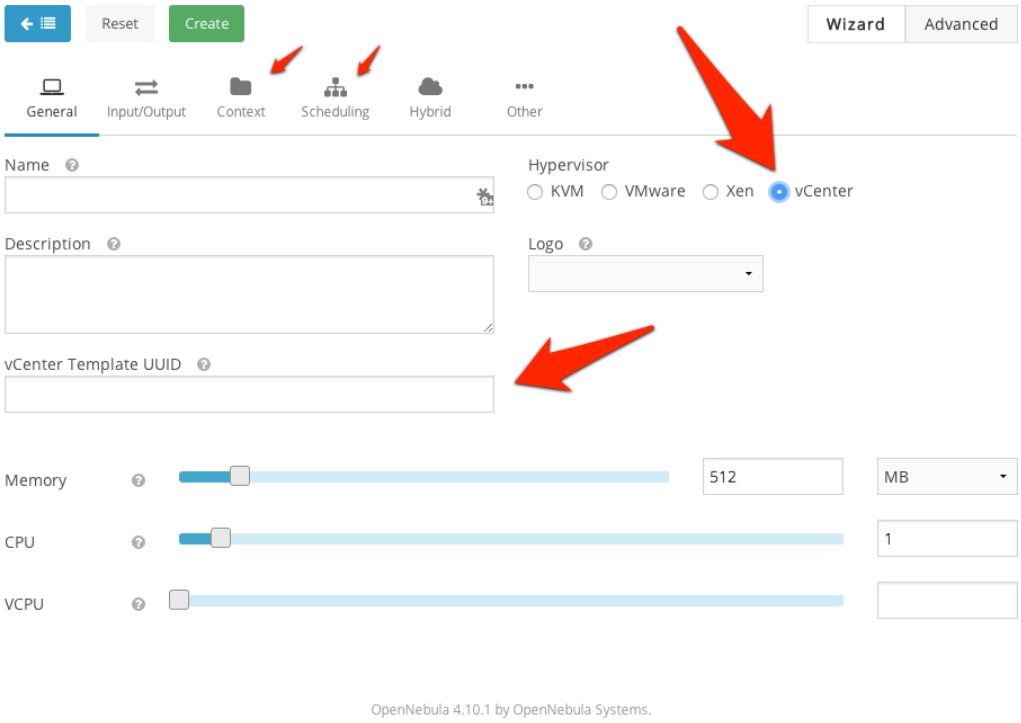
Using *the automated process for importing vCenter infrastructures*, vOneCloud will generate the above template for you at the time of importing vCenterA.

6.2.3 Add Multi Cluster VM Template

A single vOneCloud VM Template can be used to represent different vCenter VM Templates in different vCenter clusters. These multi cluster templates must be created using the Advanced tab in the VM Template creation dialog of vOneCloud, stating two or more PUBLIC_CLOUD tags, one per vCenter VM Template that wants to be referenced.

The list of attributes that can be used to create vOneCloud VM Templates through the Advanced tab follows:

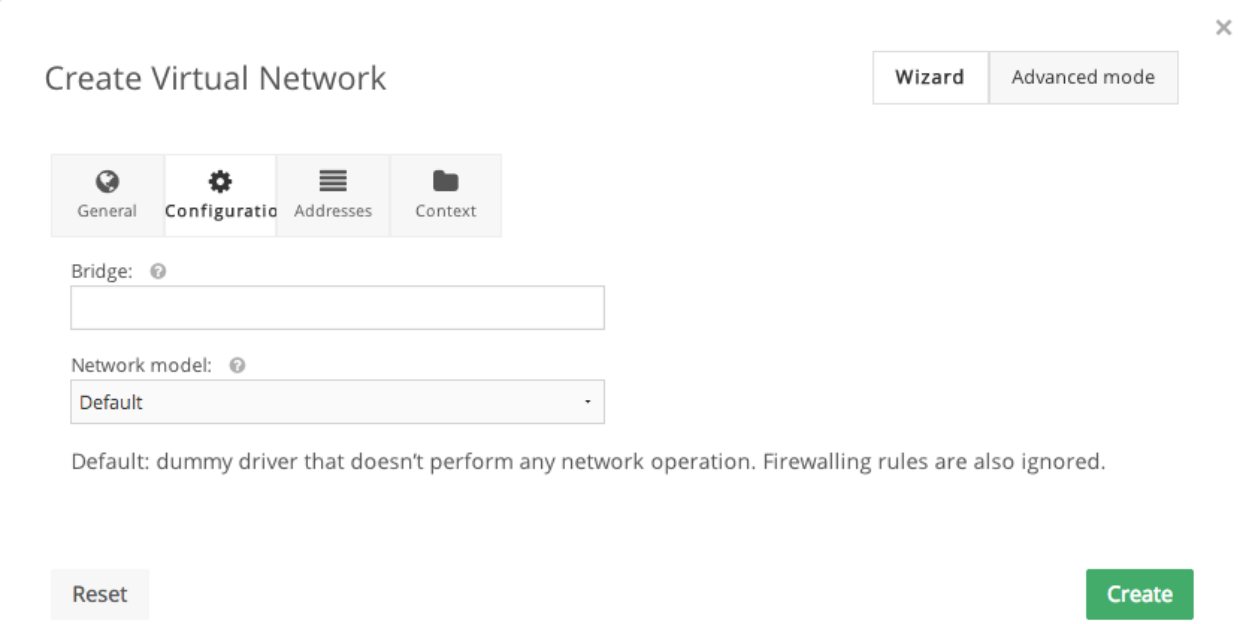
Attribute	Meaning
CPU	Physical CPUs to be used by the VM. This does not have to relate to the CPUs used by the vCenter VM Template, OpenNebula will change the value accordingly
MEMORY	Physical Memory in MB to be used by the VM. This does not have to relate to the CPUs used by the vCenter VM Template, OpenNebula will change the value accordingly
NIC	Valid MODELs are: virtuale1000, virtuale1000e, virtualpcnet32, virtualsriovethernetcard, virtualvmxnetm, virtualvmxnet2, virtualvmxnet3.
GRAPHICS	Multi-value - Only VNC supported.
PUBLIC_CLOUD	Multi-value. TYPE must be set to vcenter, VM_TEMPLATE must point to the uuid of the vCenter VM that is being represented and HOST must refer to the name of the vCenter Cluster (represented by a vOneCloud host) where the template is available
SCHED_REQUIREMENTS	Host="name of the vCenter cluster where this VM Template can instantiated into a VM".
CONTEXT	All sections will be honored except FILES
KEEP_DISKS_ON_DONE	(Optional) Prevent OpenNebula from erasing the VM disks upon reaching the done state (either via shutdown or cancel)



6.2.4 Add New Network/Distributed vSwitch

vCenter Networks/Distributed vSwitches for a particular vCenter cluster can be imported in vOneCloud after the cluster is imported using the same procedure to import vCenter clusters, making use of the `Infrastructure --> Hosts` tab in the vCenter View.

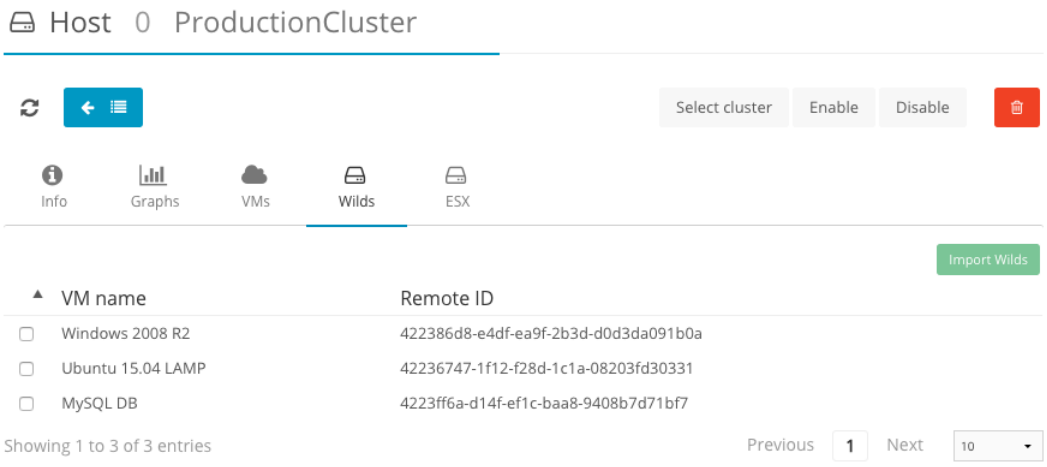
A representation of a vCenter Network or Distributed vSwitch in vOneCloud can be created in vOneCloud by creating a Virtual Network and setting the BRIDGE property to **exactly the same name as the vCenter Network**. Leave "Default" network model if you don't need to define VLANs for htis network, otherwise chose the "VMware" network model.



Several different Address Ranges can be added as well in the Virtual Network creation and/or Update dialog, pretty much in the same way as it can be done at the time of acquiring the resources explained in the *Import vCenter guide*.

6.2.5 Import Running VMs

Running and **Powered Off VMs** can be imported through the WILDS tab in the Host info tab representing the vCenter cluster where the VMs are running in.



Read more about the vCenter drivers.

6.3 Hybrid Clouds

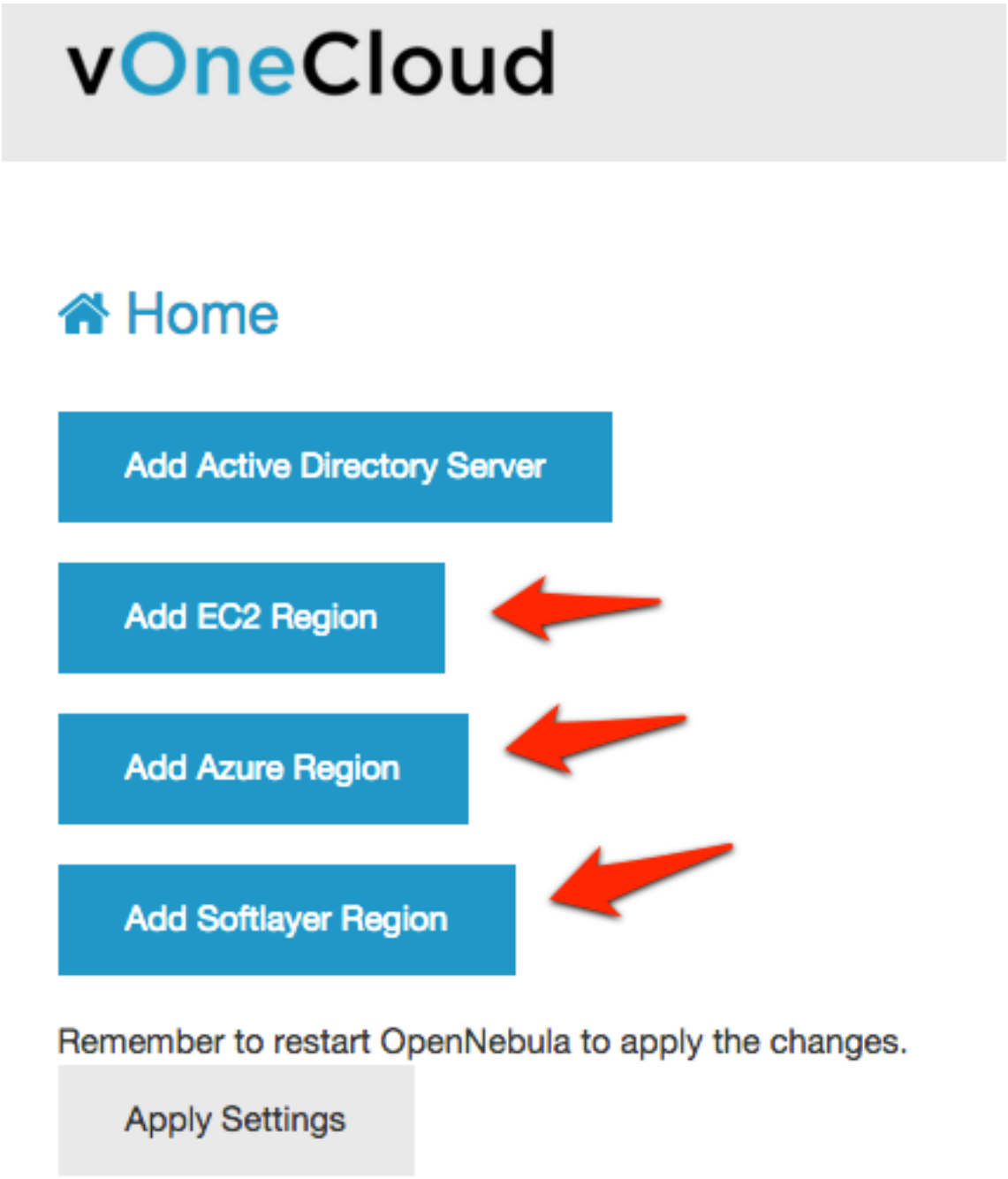
vOneCloud is capable of outsourcing virtual machines to public cloud providers. This is known as cloud bursting, and it is a feature of hybrid clouds where VMs are launched in public clouds if the local infrastructure is saturated.

If you want to extend your private cloud (formed by vOneCloud and vCenter) to create a hybrid cloud, you will need to configure at least one of the supported public clouds: Amazon EC2, IBM SoftLayer and Microsoft Azure. All hybrid drivers are already enabled in vOneCloud, but you need to configure them first with your public cloud credentials.

You will need to access the *Control Panel* in order to configure the hybrid support in vOneCloud.

6.3.1 Step 1. Configure a Hybrid Region

In the Control Panel is possible to add regions of Amazon EC2, IBM SoftLayer and Microsoft Azure to be used within vOneCloud.



Each region from the different supported cloud providers have different requirements in terms of configuration:

Amazon EC2

The capacity that you attach to this region will define the maximum number and type of Virtual Machines that vOneCloud will be able to launch in the represented Amazon EC2 region. The different instance types are defined as follows:

Name	Memory	CPU
m1.small	1.7 GB	1
m1.medium	3.75 GB	1
m1.large	7.5 GB	2

Follow the tool tips that appear on mouse over to correctly configure the parameters.

You need the Access and Secret key to be retrieved from your AWS account. More information on Amazon EC2 support can be found here.

MS Azure

The capacity that you attach to this region will define the maximum number and type of Virtual Machines that vOneCloud will be able to launch in the represented MS Azure region. The different instance types are defined as follows:

Name	Memory	CPU
Small	1.75 GB	1
Medium	3.5 GB	2
Large	7 GB	4

Follow the tool tips that appear on mouse over to correctly configure the parameters.

You need the Pem Management Certificate to be retrieved from your AWS account. Follow the next steps to craft a valid certificate:

- First, the Subscription ID, that can be uploaded and retrieved from Settings -> Subscriptions
- Second, the Management Certificate file, that can be created with the following steps. We need the .pem file (for the ruby gem) and the .cer file (to upload to Azure):

```
## Install openssl
## CentOS
$ sudo yum install openssl
## Ubuntu
$ sudo apt-get install openssl

## Create certificate
$ openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout myPrivateKey.key -out myCert.pem
$ chmod 600 myPrivateKey.key

## Concatenate key and pem certificate
$ cat myCert.pem myPrivateKey.key > vOneCloud.pem

## Generate .cer file for Azure
$ openssl  x509 -outform der -in myCert.pem -out myCert.cer
```

- Third, the certificate file (.cer) has to be uploaded to Settings -> Management Certificates

Afterwards, copy the context of the .pem certificate in the clipboard and paste it in the text area of the Control Panel Pem Management Certificate field.

More information on MS Azure support can be found here.

Note: Azure hybrid connectors only support non authenticated http proxies

IBM SoftLayer

The capacity that you attach to this region will define the maximum number and type of Virtual Machines that vOneCloud will be able to launch in the represented IBM SoftLayer region. The different instance types are defined as follows:

Name	Memory	CPU
slcci.small	1 GB	1
slcci.medium	4 GB	2
slcci.large	8 GB	4

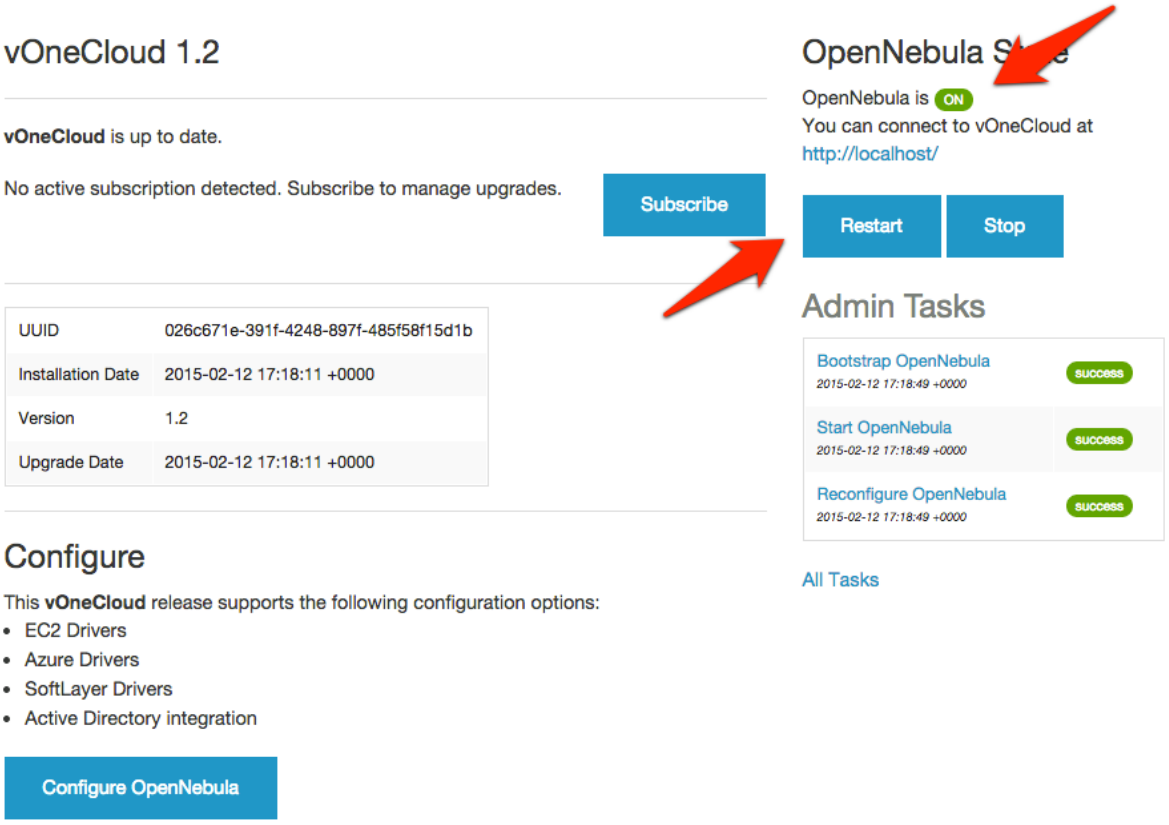
Follow the tool tips that appear on mouse over to correctly configure the parameters.

You need your SoftLayer Username and the API Key that can be retrieved from your SoftLayer Control Panel. More information on IBM SoftLayer support can be found here.

Warning: If vOneCloud is running behind a corporate http proxy, the SoftLayer hybrid connectors won't be available

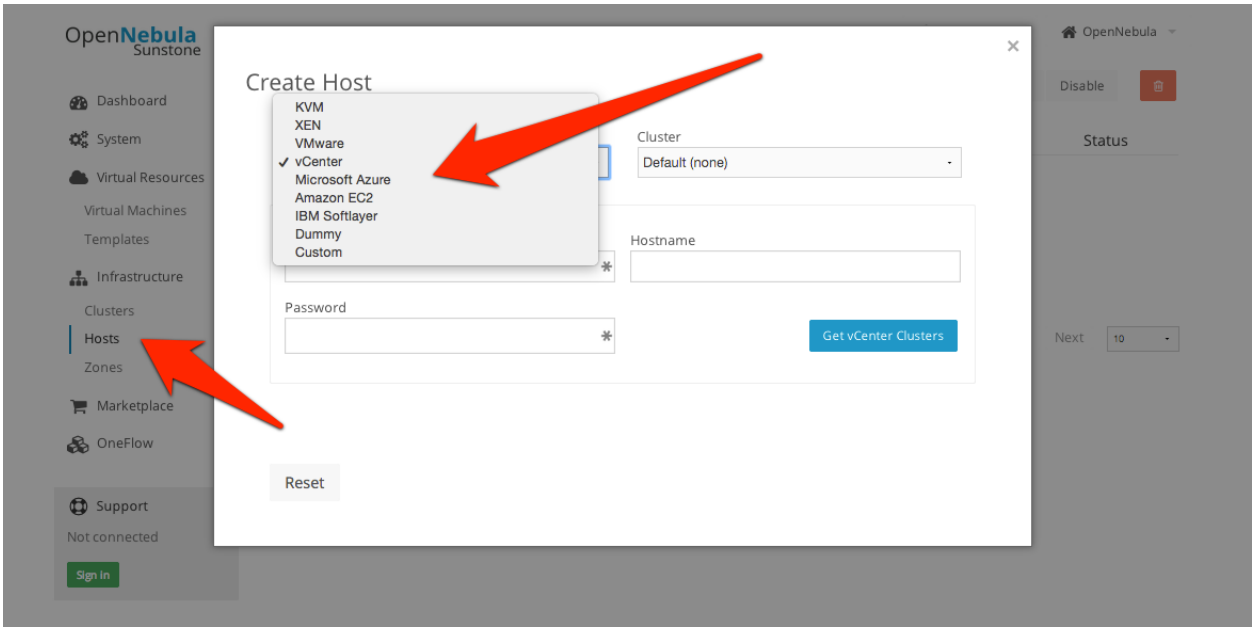
6.3.2 Step 2. Restart vOneCloud services

Click on the "Apply Settings" button. For changes to take effect, you need to restart vOneCloud services and wait for OpenNebula state to be ON.



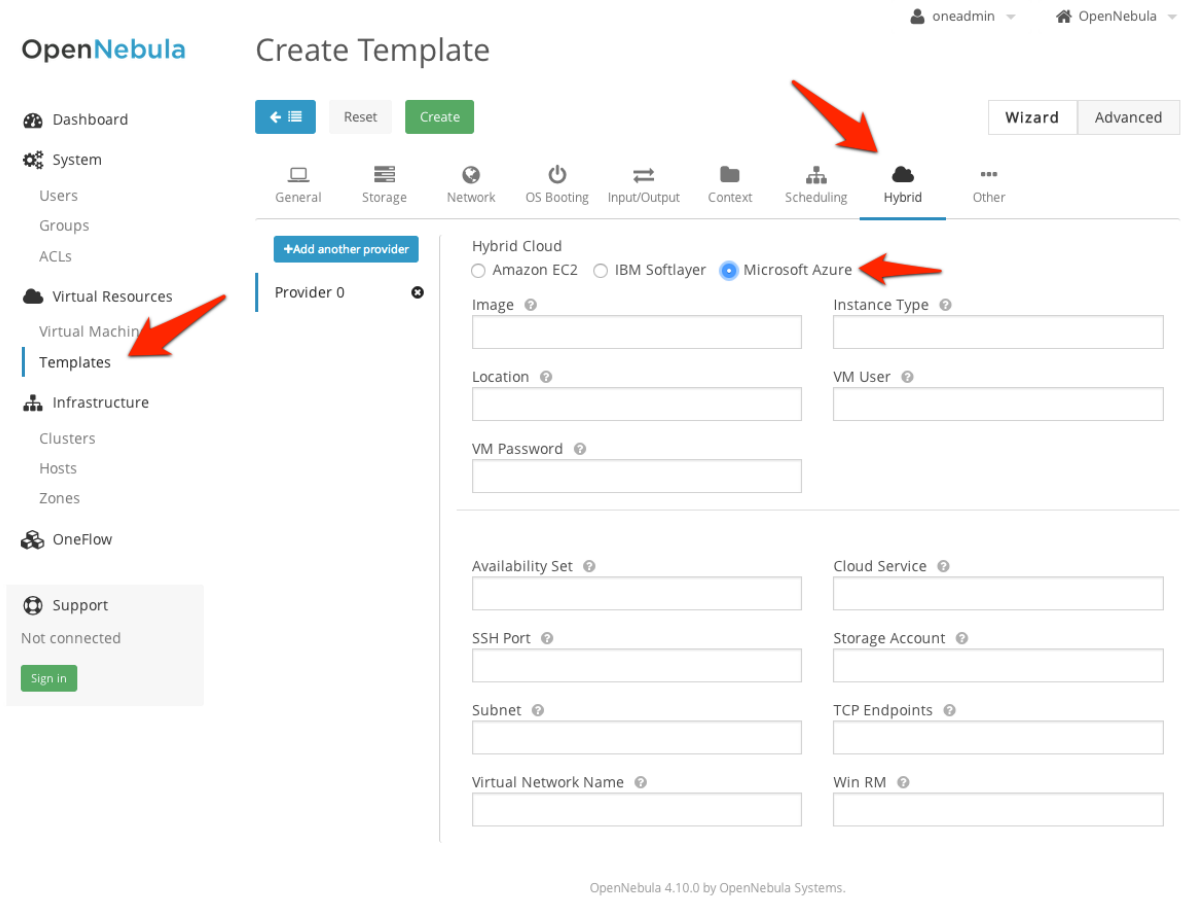
6.3.3 Step 3. Create vOneCloud hybrid resources

Afterwards, each region can be represented by vOneCloud hosts can be added from the vCenter View:

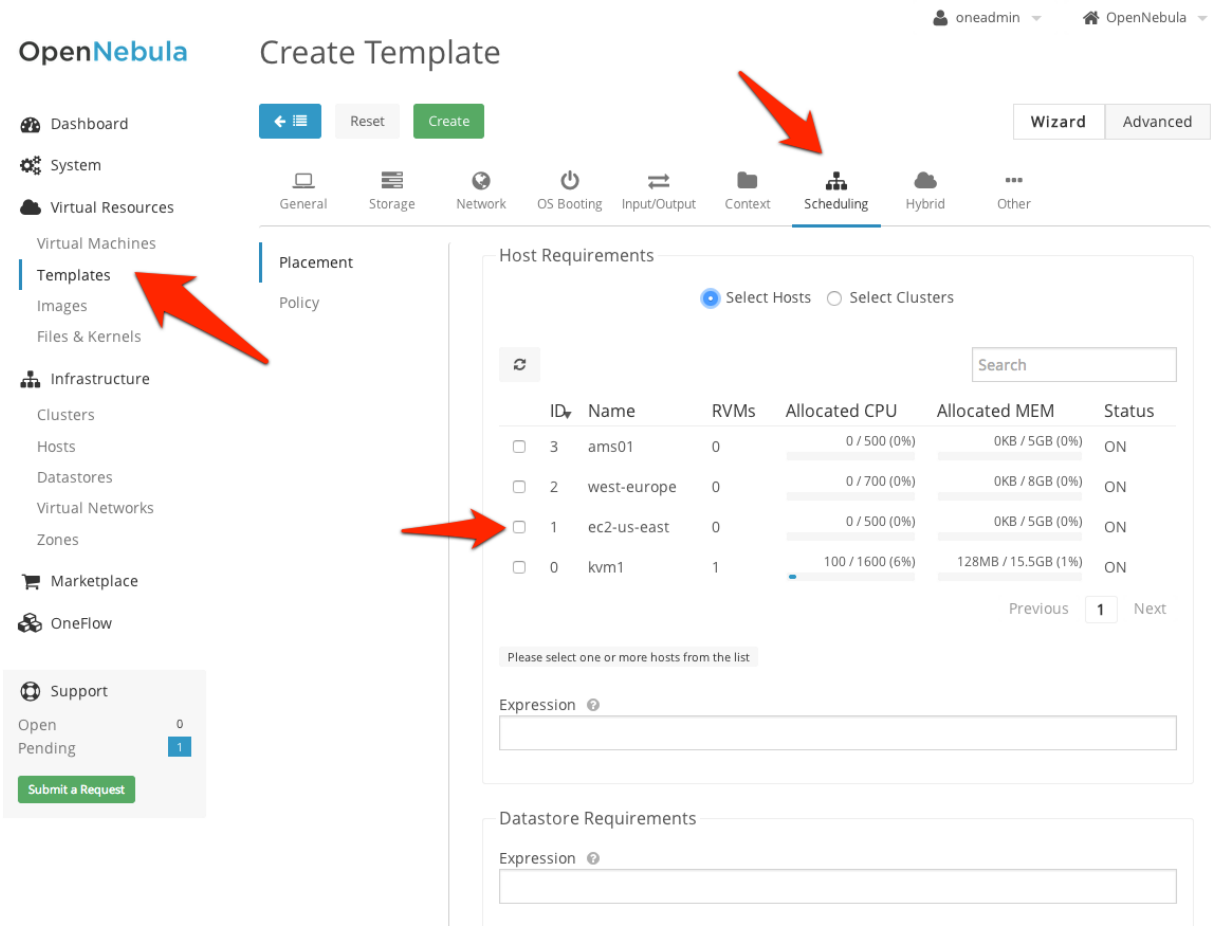


The hybrid approach is carried out using hybrid templates, which represents the virtual machines locally and remotely.

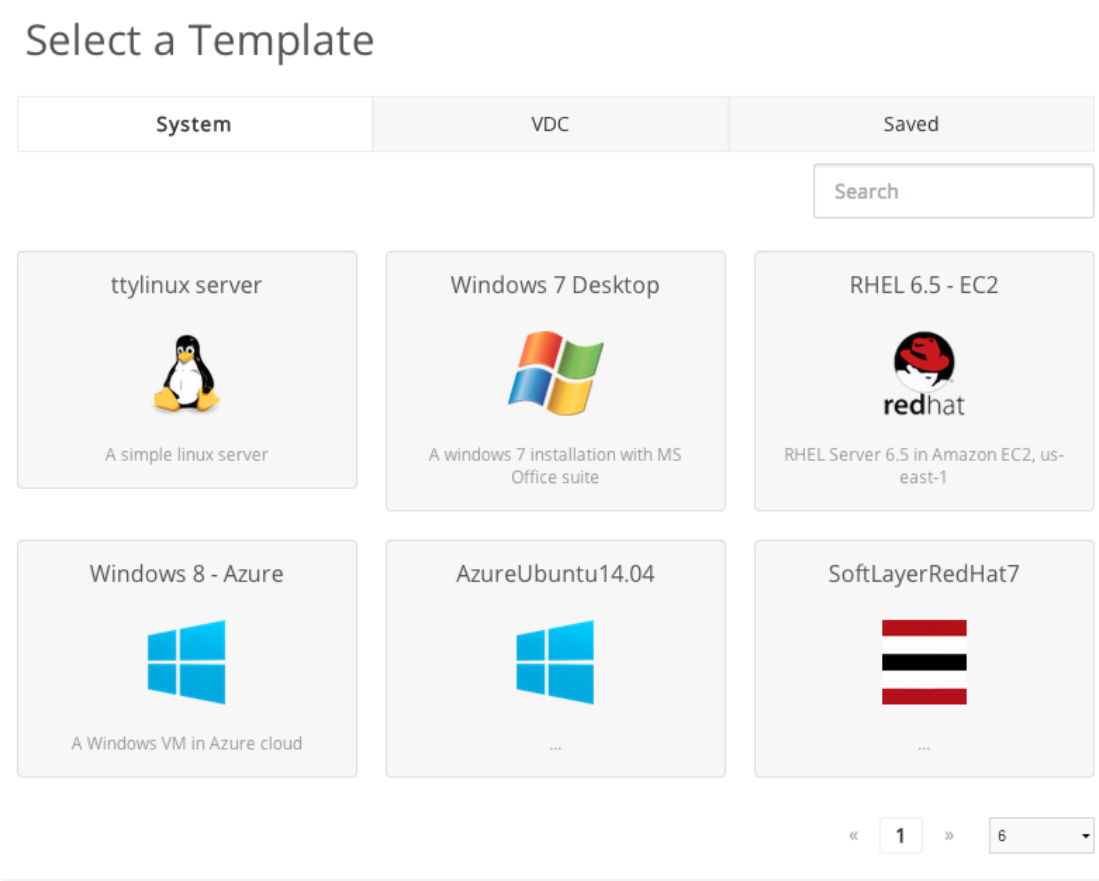
The idea is to build a vOneCloud hybrid VM template that represents the same VM in vCenter and in the public cloud. This can be carried out using the hybrid section of the VM Template creation dialog (you can add one or more public cloud provider)



Moreover, you need to add in the Scheduling tab a proper host representing the appropriate public cloud provider. For instance, for an EC2 hybrid VM Template:



Once templates are ready, they can be consumed at VM creation time from the Cloud View:

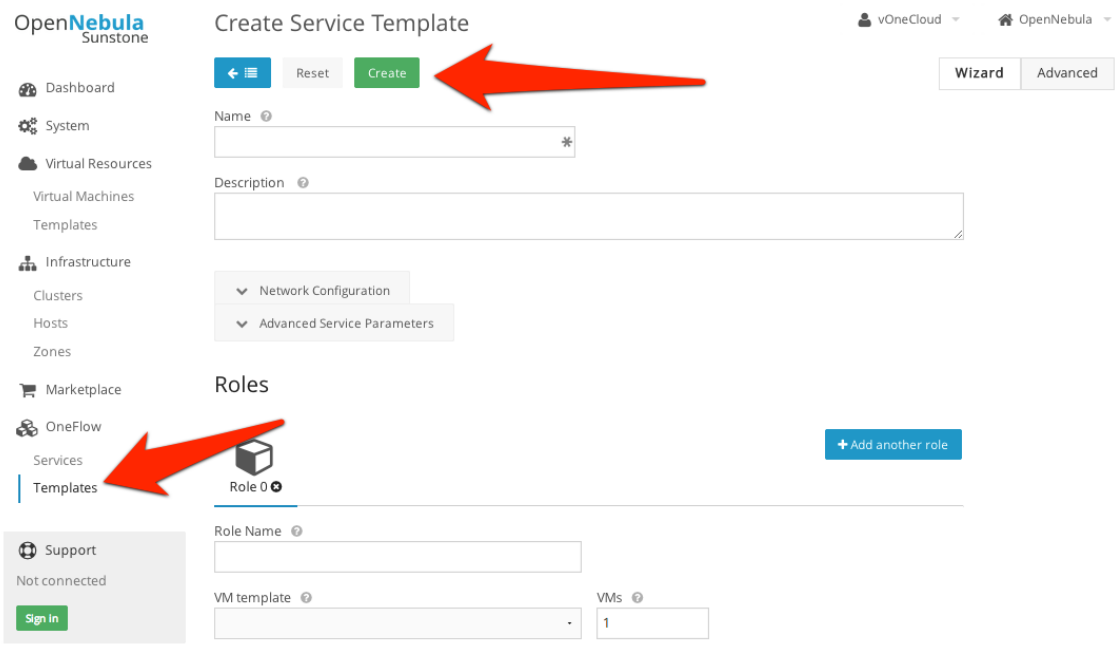


Learn more about hybrid support.

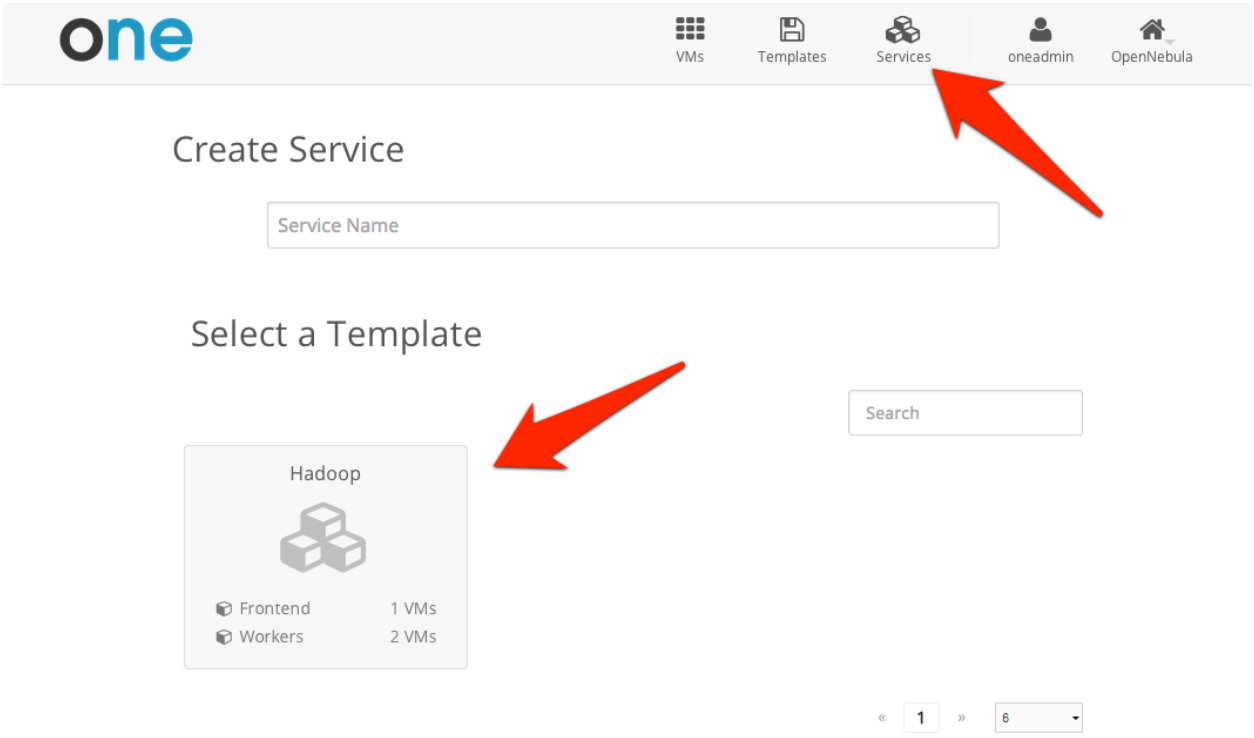
6.4 Multi VM Applications

vOneCloud enables the management of individual VMs, but also the management of sets of VMs (services) through the OneFlow component.

vOneCloud ships with a running OneFlow, ready to manage services, allowing administrators to define multi-tiered applications using the vCenter View:



End users can consume services from the Cloud View:



Elasticity of each service can be defined in relation with chosen Key Performance Indicators as reported by the hypervisor.

Note: vOneCloud does not include the `onegate` component which is mentioned at some places in the application flow guide.

More information on this component in the OneFlow guide. Also, extended information on how to manage multi-tier

applications is available this guide.

6.5 Authentication

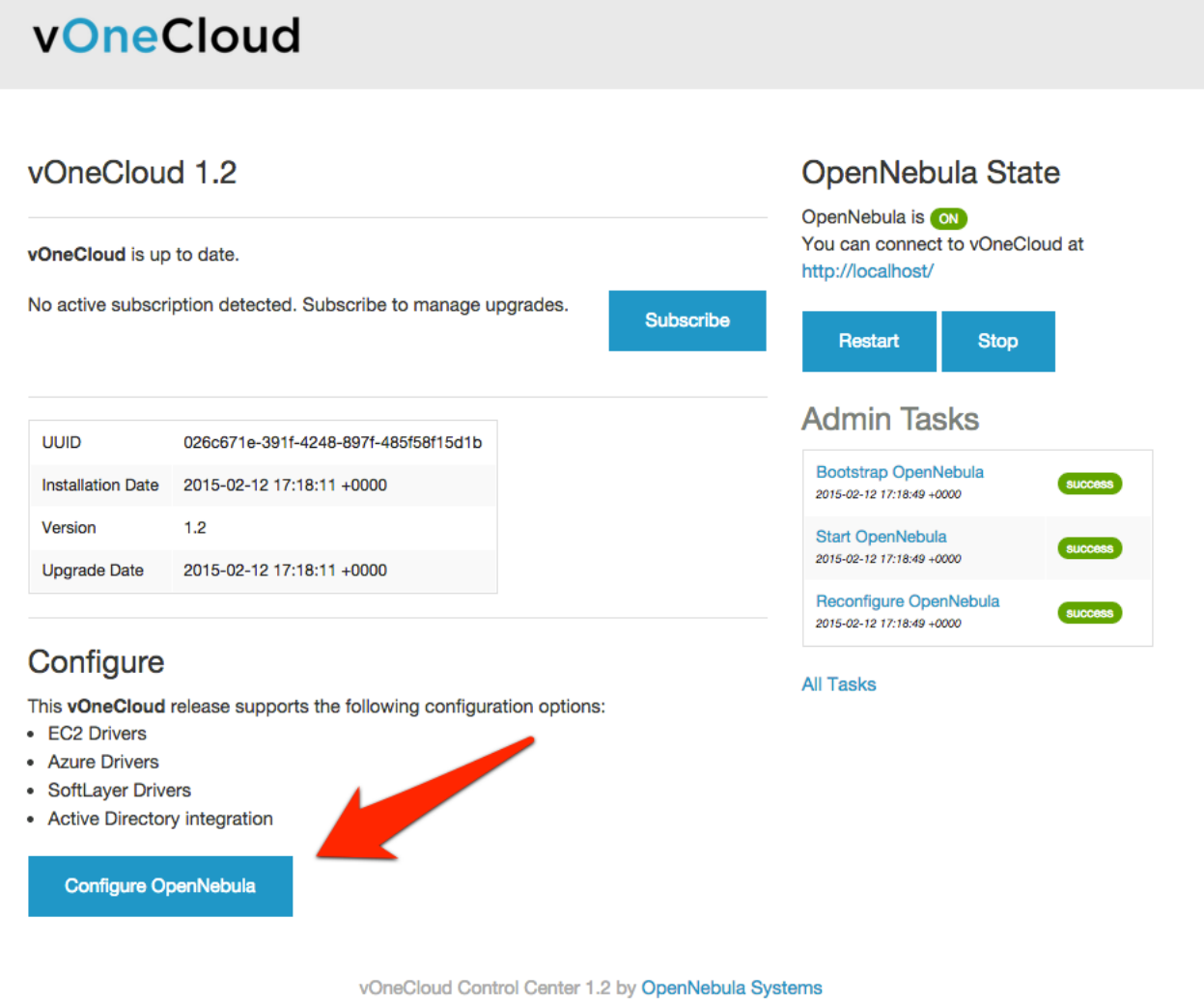
By default, vOneCloud authentication uses an internal user/password system with user and group information stored in an internal database.

vOneCloud can pull users from a corporate Active Directory (or LDAP), all the needed components are enabled and just an extra configuration step is needed. As requirements, you will need an Active Directory server with support for simple user/password authentication, as well as a user with read permissions in the Active Directory user's tree.

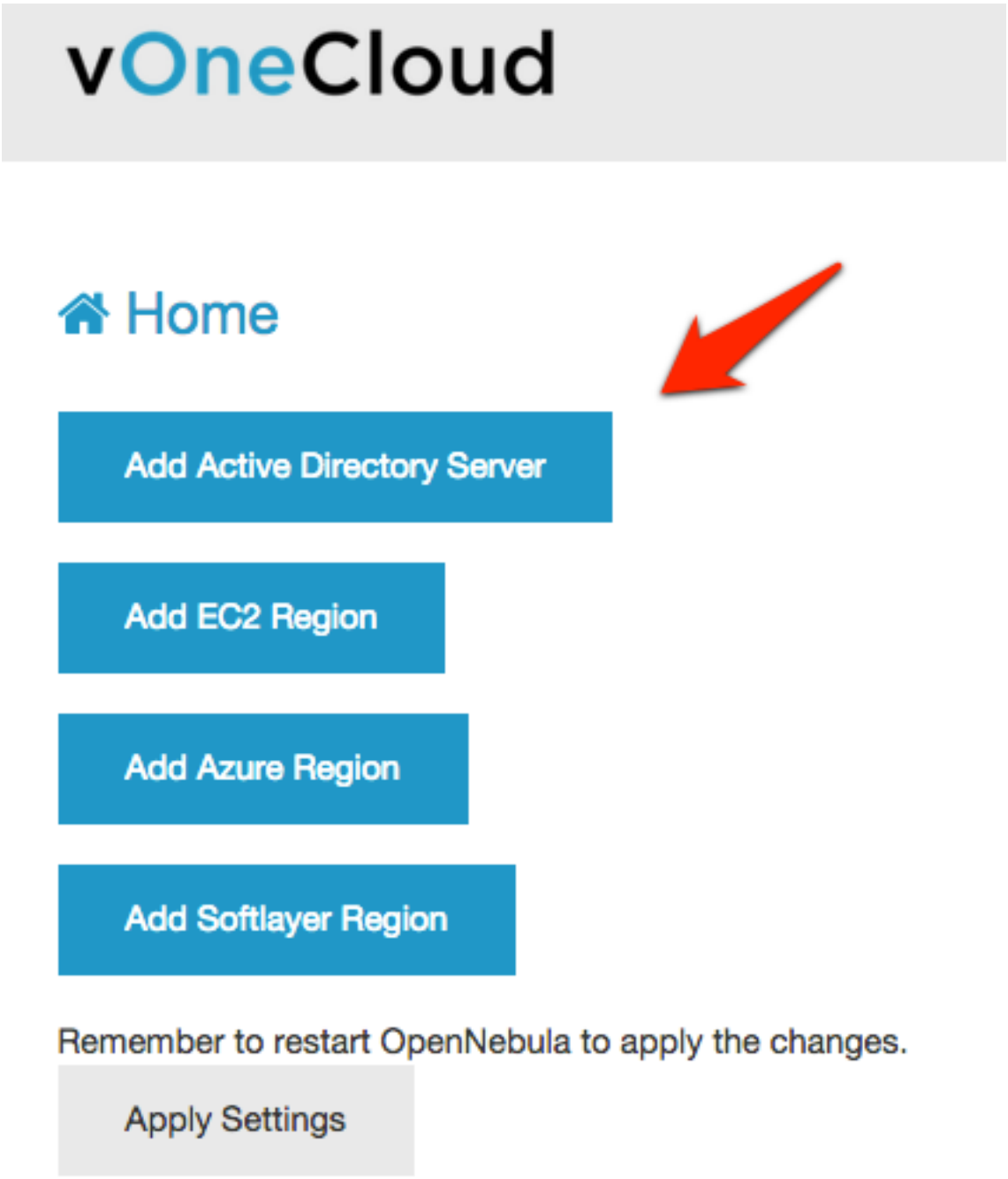
You will need to access the *Control Panel* in order to configure the Active Directory support in vOneCloud. After the configuration is done, users that exist in Active Directory can begin using vOneCloud.

6.5.1 Step 1. Configure Active Directory support

Click on the "Configure OpenNebula" button

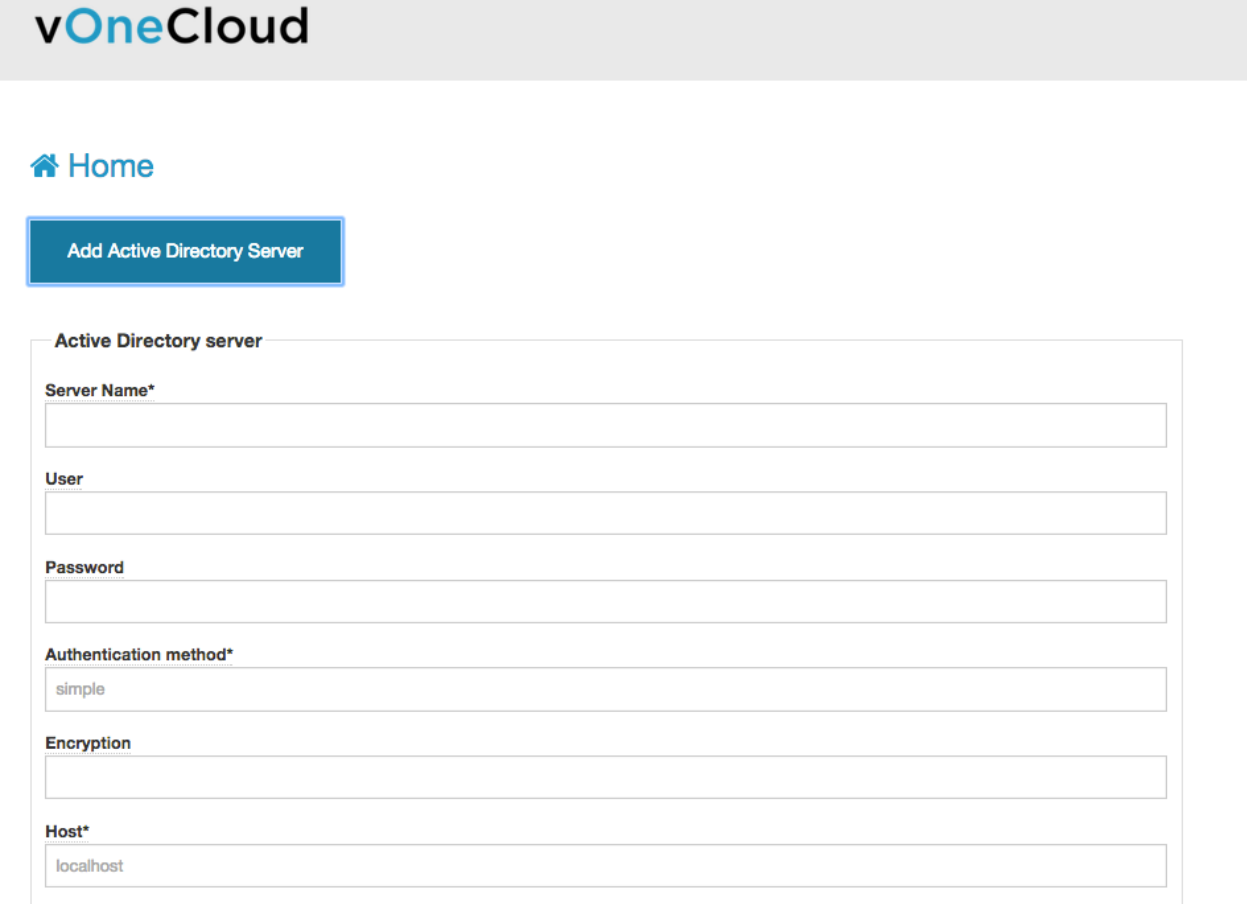


In the following screen, select the "Add Active Directory" category



Fill the needed fields following the criteria described in the next table

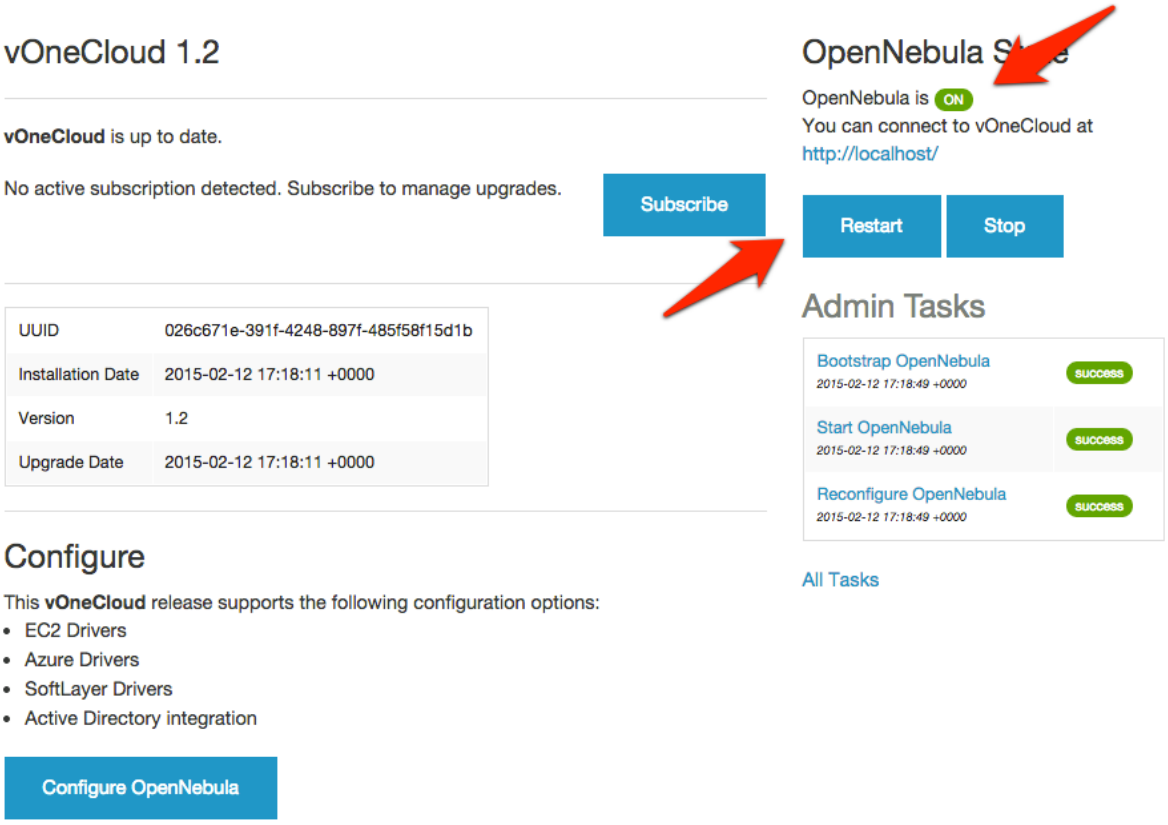
Attribute	Description
Server Name	Chosen name for the authentication backend
User	Active Directory user with read permissions in the user's tree plus the domain.
Password	Active Directory user password
Authentication method	Active Directory server authentication method (eg simple)
Encryption	simple or simple_tls
Host	hostname or IP of the Domain Controller
Port	port of the Domain Controller
Base Domain	base hierarchy where to search for users and groups
Group	group the users need to belong to. If not set any user will do
User Field	Should use **sAMAccountName** for Active Directory. Holds the user name, if not set 'cn' will be used
Group Field	field name for group membership, by default it is 'member'
User Group Field	user field that that is in in the group group_field, if not set 'dn' will be used



Click on the "Apply Settings" button when done.

6.5.2 Step 2. Restart vOneCloud services

For changes to take effect, you need to restart vOneCloud services and wait for OpenNebula state to be ON.



You can find more infromation on the integration with Active Directory in this guide.

vOneCloud supports are a variety of other authentication methods with advanced configuration, follow the links to find the configuration steps needed (*Advanced Login* needed):

X509 Authentication	Strengthen your cloud infrastructure security
SSH Authentication	Users will generate login tokens based on standard ssh rsa keypairs for authentication

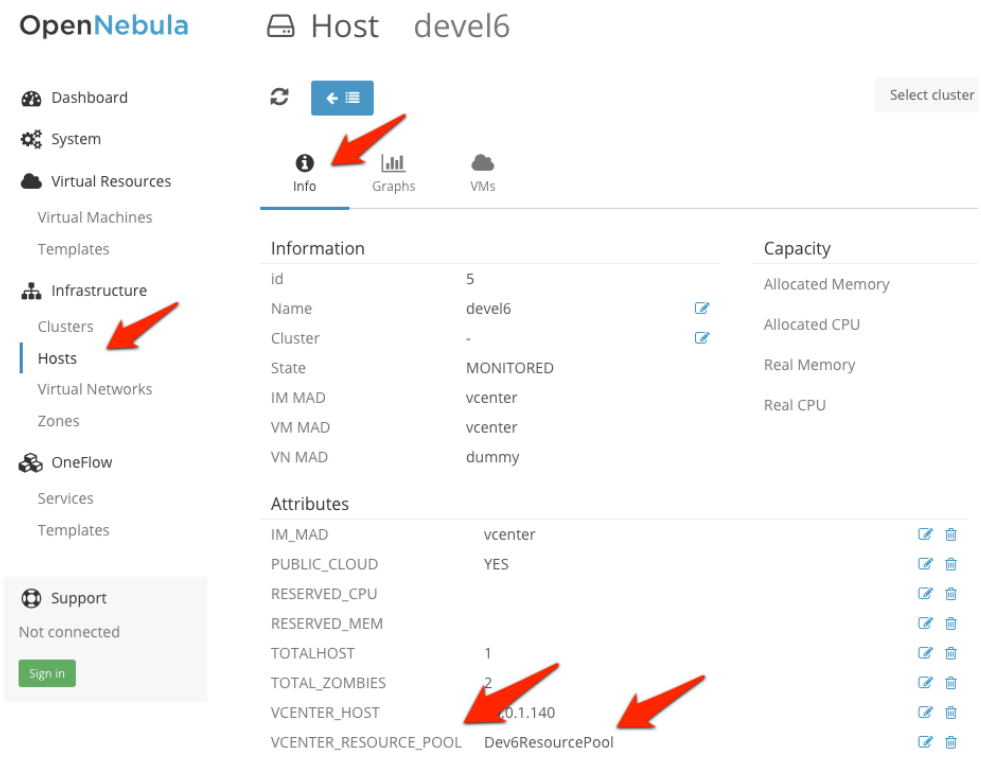
6.6 Resource Pool Confinment

vCenter credentials of the user that vOneCloud is configured with to access the vCenter clusters can be confined into a Resource Pool, to allow only a fraction of the vCenter infrastructure to be used by vOneCloud users. The steps to confine vOneCloud users into a Resource Pool are:

- Create a new vCenter user
- Create a Resource Pool in vCenter and assign the subset of Datacenter hardware resources wanted to be exposed through vOneCloud
- Give vCenter user Resource Pool Administration rights over the Resource Pool
- Give vCenter user Resource Pool Administration (or equivalent) over the Datastores the VMs are going to be running on

Afterwards, these credentials can be used to add to vOneCloud the host representing the vCenter cluster. Add a new tag called VCENTER_RESOURCE_POOL to the host template representing the vCenter cluster (for instance, in the info tab of the host, or in the CLI), with the name of the Resource Pool. All the VMs created through vOneCloud would be confined to this Resource Pool.

Note: Remember to modify the VCENTER_PASSWORD tag as well, since after the VCENTER_RESOURCE_POOL update it will get double encrypted. This *limitation* will be addressed in future releases of vOneCloud.



CHAPTER

SEVEN

APPLIANCE CONFIGURATION

7.1 Introduction

The vOneCloud appliance features two components to simplify the configuration tasks needed to set-up, configure, maintain and upgrade the cloud: the vOneCloud Control Console (text-based) and the vOneCloud Control Panel (web-based).

This sections explains each of these interfaces, how to access them and the available configuration options.

7.2 Control Console

This is a text-based interface available used to run basic configuration tasks in the vOneCloud appliance.

The Control Console is available by opening the vOneCloud appliance console in vCenter. It requires no authentication since only the vCenter administrator will be able to open the vOneCloud console.

This component runs in two stages. The initial bootstrap stage, and the basic configuration stage.

7.2.1 Initial Boostrap

The initial bootstrap is a configuration wizard which is part of the deployment process of vOneCloud, and it **must** be run. During this step the user will be prompted to configure the following aspects:

- Configure Network
- Set the root password
- Change the password for oneadmin in OpenNebula
- Configure http proxy

Once this wizard has been executed the user is ready to open the vOneCloud Control Panel at *http://<appliance_ip>:8000* in order to continue with the deployment configuration and to start the OpenNebula service.

Note that during this step the *oneadmin* account password will be set, which will be then used to access the vOneCloud Control Panel.

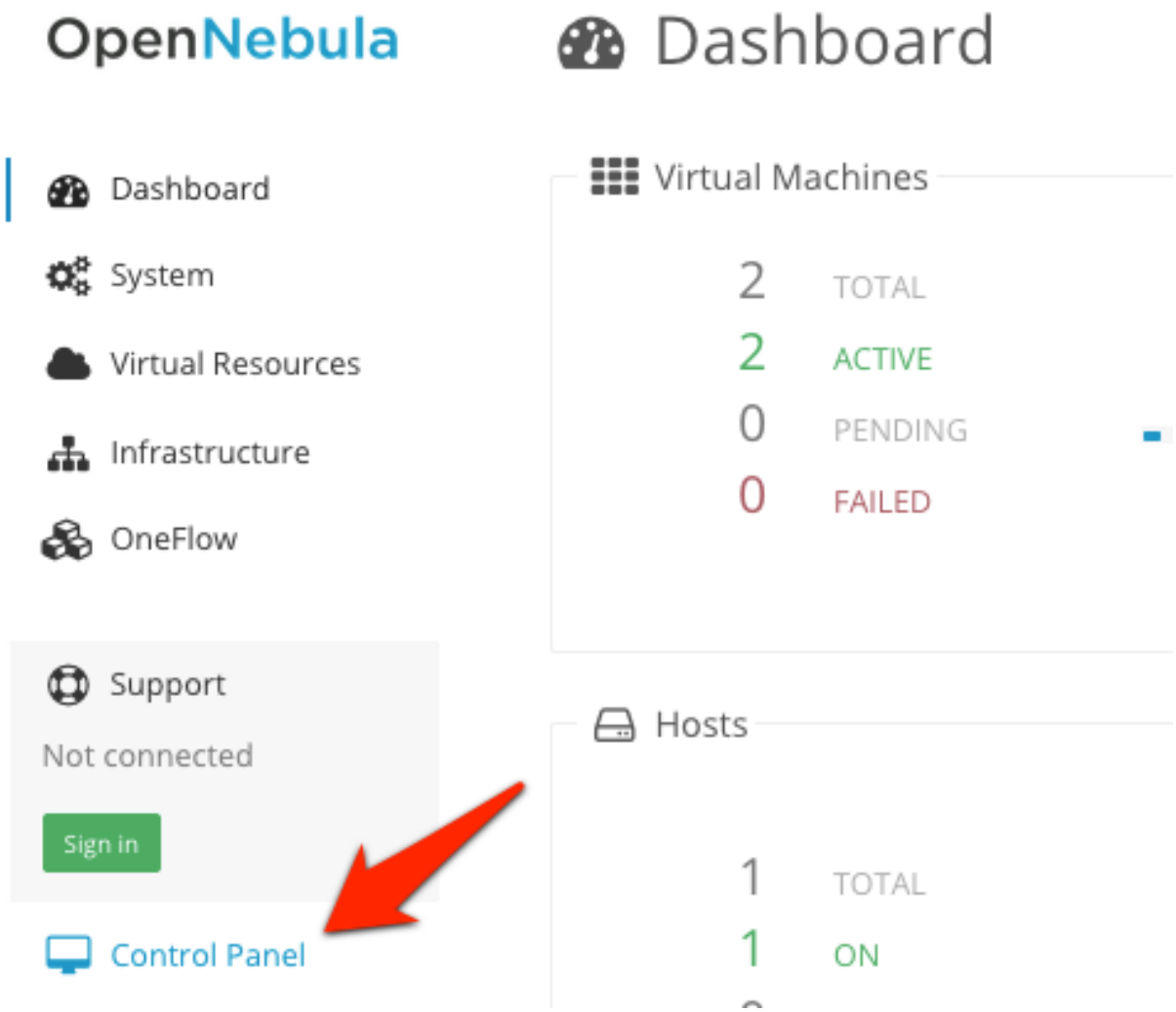
7.2.2 Basic Configuration

At any given moment, the vOneCloud administrator may choose to open the vOneCloud appliance console in vCenter to perform some additional configuration:

- Networking configuration, which is useful if the networking configuration changes at any given time.
- Proxy configuration.
- Change the oneadmin password. Note that this step requires that the vOneCloud administrator restarts the OpenNebula service in the *vOneCloud Control Panel*.

7.3 Control Panel

This is a web based interface available at *http://<appliance_ip>:8000* which handles many aspects of the vOneCloud platform configuration. The Control Panel can be reached at any time from the Sunstone GUI using the Control Panel link in the bottom of the left hand side menu.



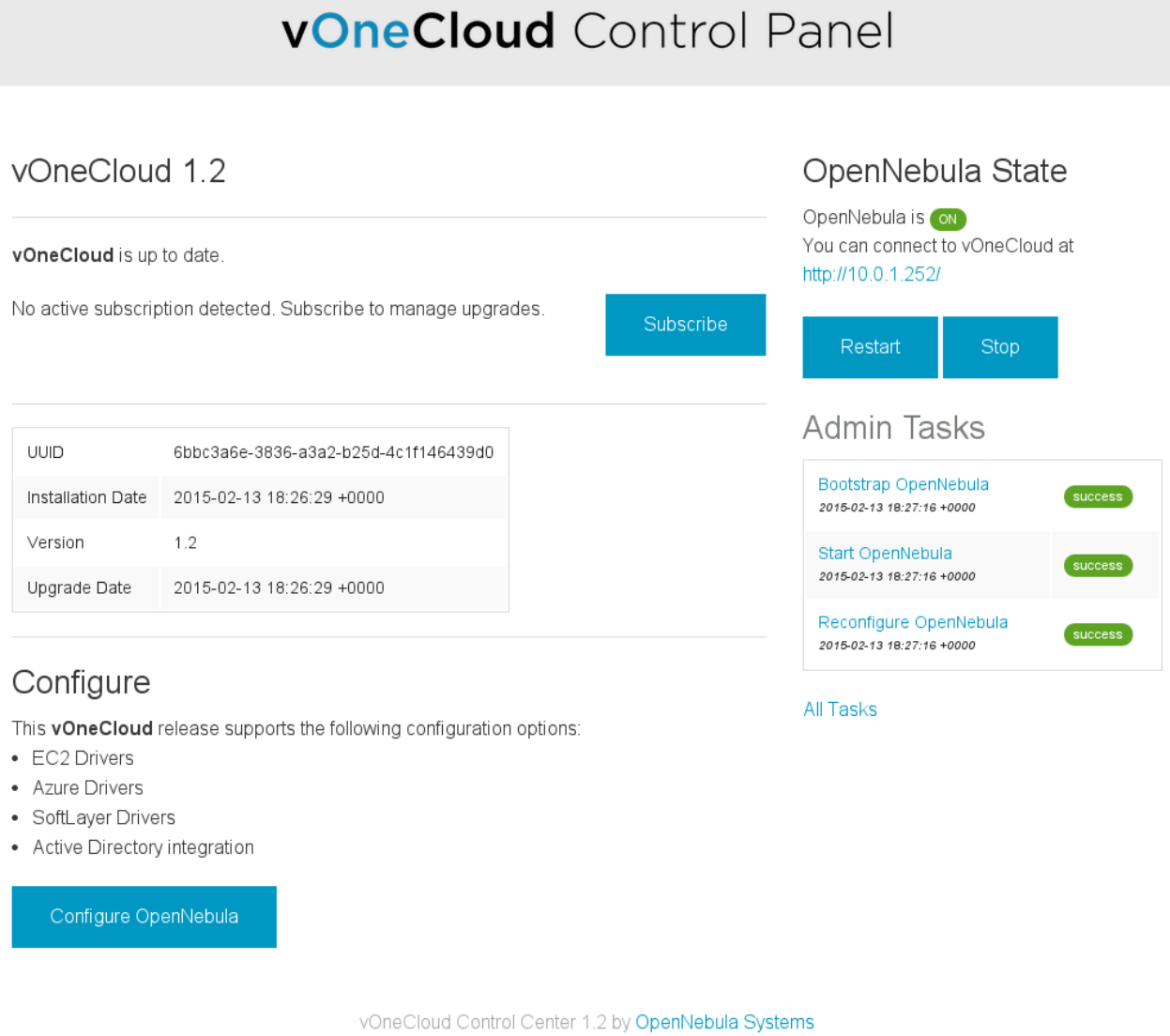
To log in the administrator will need the *oneadmin* account, which is set in the initial configuration of the Control Console.

The next section documents the available information and actions in this interface

7.3.1 Appliance Management

In the dashboard of the Control Panel you will be able to see the following information:

Parameter	Description
UUID	Each vOneCloud appliance has an automatically generated UUID used to identify it. This information is required by vOneCloud Support for users with an active support subscription.
Installation Date	Records the date of the vOneCloud first deployment.
Version	Active vOneCloud version
Upgrade Date	Records the date of last vOneCloud upgrade.



Additionally vOneCloud will report the subscription status:

- No subscription detected
- Active subscription
- Expired subscription

7.3.2 Configuration Management

The configuration action handles the supported configuration of the vOneCloud appliance:

- *Hybrid drivers (Amazon EC2, IBM SoftLayer, MS Azure)*.
- *Active Directory or LDAP integration*.
- *System Options - Enable SSH*.
- *System Options - Enable SSL*.

If the configuration is changed while OpenNebula is running, it will need to be restarted. A warning will appear in the dashboard reminding the user to restart the OpenNebula service.

7.3.3 System options

It is possible to configure SSH and SSL:

SSH

By default SSH access is disabled. If you want to enable it, simply enable the checbock.

SSL

If you want to enable SSL you will need to:

- Enable the *SSL enabled* checkbox
- Provide a Certificate (copy&paste the contents of the file)
- Provide a Key Certificate (copy&paste the contents of the file)
- Optionally, provide the CA Certificate (copy&paste the contents of the file)

7.3.4 Service Management

The OpenNebula services can be managed in the main dashboard: start, stop and restart.

Any of this actions will trigger one or more tasks. If one of this tasks fails, the user will be notified, and those with an active support subscription will be able to send the error report to the vOneCloud Support.

7.3.5 Log Access

The Control Panel features the possibility to access the OpenNebula logs.

7.3.6 Automatic Upgrades

When a new vOneCloud release is available for download users will be notified. User with an active support subscription will be able to upgrade with a single click. In the main Dashboard area the user will be notified if there is a new release available. In that case the user will be able to click a button that will start the upgrade.

Note: Before running an automatic upgrade users are recommend to create a vCenter snapshot of the vOneCloud appliance in order to revert back to it in case of failure.

7.4 Advanced Customizations

This section documents further customizations that can tailor the vOneCloud environment to your needs. However, these modifications will be lost after an upgrade. So please document the process exactly so you can replay it after upgrading the appliance.

All the customizations documented in this section require logging into the vOneCloud appliance, see the *Logging into the Appliance* guide to access it.

These customizations are currently supported:

- *Rebrand vOneCloud*

Warning: The following changes will be lost after an upgrade.

7.4.1 Rebranding

It is possible to change the logos of the Sunstone interface by replacing these files:

- Logo for the login screen: `/usr/lib/one/sunstone/public/images/opennebula-sunstone-v4.0.png`. The original size is 355 x 78 px. The image will be force resized to a width of 355px.
- Logo for the admin view: `/usr/lib/one/sunstone/public/images/opennebula-sunstone-v4.0-small.pn` The original size is 413 x 60 px.
- Logo for the group and cloud view: `/usr/lib/one/sunstone/public/images/one_small_logo.png`. The original size is 563 x 194px.

The background of the login screen can be customized by replacing `/usr/lib/one/sunstone/views/login.erb`, with these contents:

```
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <link rel="shortcut icon" href="images/favicon.ico" />
    <title>OpenNebula Sunstone Login</title>
    <link rel="stylesheet" type="text/css" href="css/login.css" />
    <!--[if IE]>
    <script type="text/javascript" src="vendor/crypto-js/core-min.js"></script>
    <script type="text/javascript" src="vendor/crypto-js/enc-base64-min.js"></script>
    <![endif]-->

    <% if $conf[:env] == 'dev' %>
      <script src="bower_components/requirejs/require.js" data-main="app/login"></script>
    <% else %>
```

```
      <script src="dist/login.js"></script>
    <% end %>
  </head>

  <body style="
  background: url(images/%YOURIMAGE%) no-repeat center center fixed;
  -webkit-background-size: cover;
  -moz-background-size: cover;
  -o-background-size: cover;
  background-size: cover;
  ">

    <% if settings.config[:auth] == "x509" %>
    <%= erb :_login_x509 %>
    <% else %>
    <%= erb :_login_standard %>
    <% end %>

    <div id="footer" style="overflow:visible;">
      <a href="http://opennebula.org" target="_blank">OpenNebula 4.13.85</a>
      by
      <a href="http://opennebula.systems" target="_blank">OpenNebula Systems</a>
      .
    </div>
  </body>
</html>
```

Make sure you replace `%YOURIMAGE%` in the above example with the name of your background. Upload your background image to `/usr/lib/one/sunstone/public/images/`. For example, if we have a logo called `server.jpg`, that line should read:

> background: url(images/server.jpg) no-repeat center center fixed;

And we should upload it to `/usr/lib/one/sunstone/public/images/server.jpg`.

After any of these changes it's necessary to restart OpenNebula in the Control Panel.

7.5 Troubleshooting

This section details what actions to take if any of the vOneCloud appliance configuration functions fails.

7.5.1 Cannot Check for Upgrades

When the vOneCloud Repository cannot be reached this message will be displayed:

> OpenNebula Systems vOneCloud Repository is unreachable. Cannot check for upgrades. Read the Troubleshooting guide for more info.

This means that the appliance cannot reach the appliance repository at vonecloud.com. In the first place, check from your browser that this website is up: https://downloads.vonecloud.com/version, it should display a message like:

> {"error":"Invalid Data."}

If that works, then it's probably a networking configuration error. Make sure that the network of the appliance has been properly set (see *here*). It also might be a proxy problem if the appliance requires a proxy to access the internet. If you are sure these configuration parameters are correct, perform a *manual login to the appliance* and check the following items:

- Inspect the routes *ip route*
- If you are not using a proxy, make sure you can reach the Google DNS to test internet connection: *ping 8.8.8.8*.
- Run the following command: *curl -kv https://downloads.vonecloud.com/version*. If you are using a proxy run this instead: *HTTPS_PROXY=http://<proxy_user>:<proxy_pass>@<proxy_host>:<proxy_port> curl -kv https://downloads.vonecloud.com/version*.

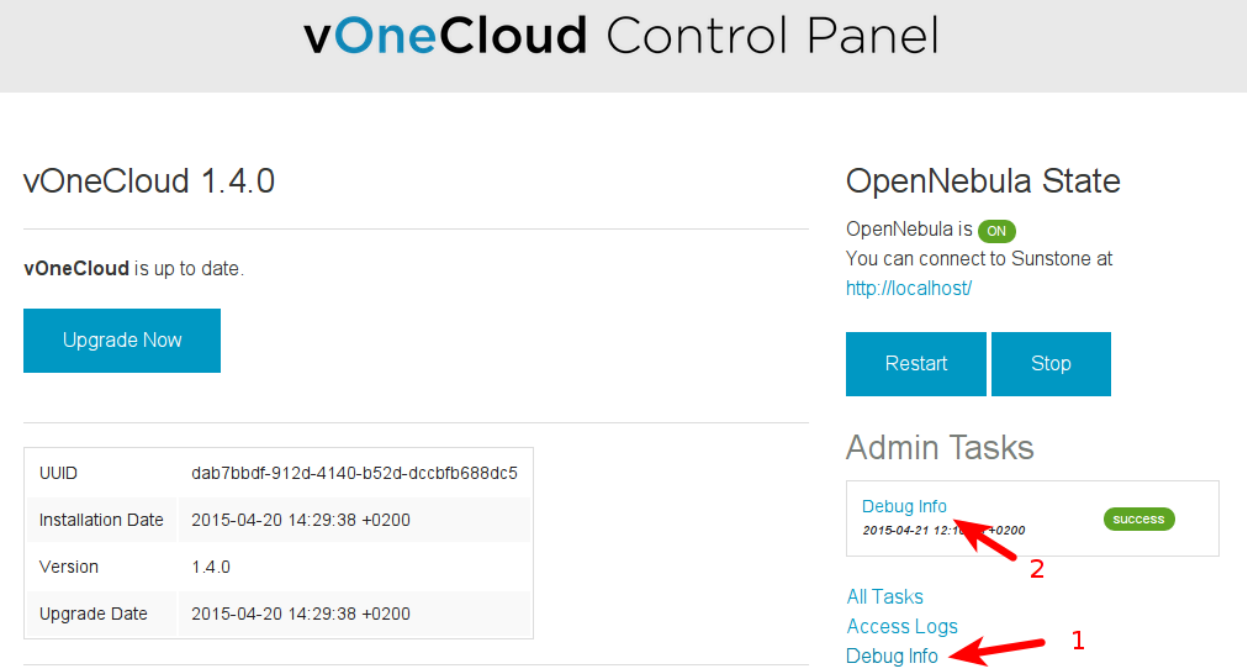
If you are sure the network is properly configured, please feel free to submit a support to vOneCloud Support.

7.5.2 Debug Information

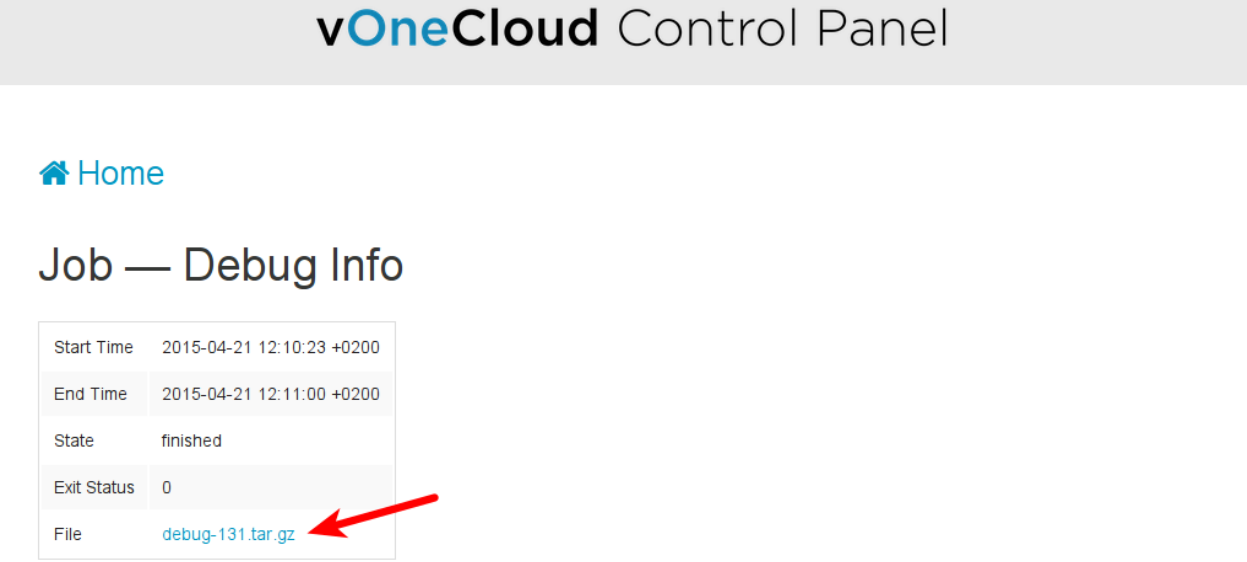
An Admin Task called `Debug Info` generates a gzipped tar file which can be downloaded that contains all the required information to debug the cloud if the OpenNebula user runs into a problem. This file can be then sent to vOneCloud Support. Note that this sends information on all the resources of the cloud and the OpenNebula log.

Note: Please examine this information before sending it over if you have concerns about sensitive data that might be automatically bundled in the file.

To generate the debug information follow these steps:



To download the file click on the `Debug Info` job and download the file:



7.5.3 Job Failure

A job should never fail. If it fails you should submit a support ticket with the attached Job Crashed Report (link found in the Job page) to vOneCloud Support.