vOneCloud Documentation

Release 3.0.0

OpenNebula Systems

Aug 03, 2017

CONTENTS

CHAPTER

ONE

RELEASE NOTES VONECLOUD 3.0

1.1 What's New vOneCloud 3.0

vOneCloud 3.0 is powered by OpenNebula 5.4.0 'Medusa', and, as such, includes functionality present in Medusa relevant to vOneCloud:

- **Enhanced storage management**, vOneCloud is fully aware of all *VMs disks*. Non-persistent images and volatiles disks are now supported.
- **Storage quotas and datastore capacity check**, never run out of capacity correctly dimensioning the available datastores and the storage *quotas* given to end users
- **Balance storage load** of *VMs across datastores* automatically without the need of Storage DRS
- **Linked clone support**, add support for *linked clones* for VMs at the time of importing a VM Template
- **Disk resize capabilities**, *resize* the capacity of a VM disk at boot time or when the VM is in poweroff
- **Save disk functionality**, *register any VM disk as an image* for later use in VMs, either directly from a VM Template or through the disk attach operation
- **Save as Template functionality**, *save any VM as a VM Template* at any point during its lifecycle
- **Folder management**, define in which *vCenter folder* you want your VMs into, on a VM to VM basis.
- **Network creation support**, a new vCenter network model is available in virtual network definition, *standard and different port groups and vSwitches can be created* from within OpenNebula. VLAN IDs, MTUs and number of ports can be specified when a port group is created.
- **Full storage and networking support** in *imported VM Templates*, images and networks representing disks and network interfaces are created for VM templates and folder placement features.
- **Improved CDROM management**, now a new *CDROM drive* is added to the VM if not present when an ISO image is attached
- **Imported VMs improvements**, with the possibility of *adding VNC to any imported VM*.
- **Removed naming limitations**, vCenter cluster and *datastore* names with spaces are now supported
- **Improved performance**, up to two orders of magnitude of speedup in monitoring and import times, as well as less error prone synchronous calls to vCenter
- **Faster VM deployment**, with up to 10 VM spinning up simultaneously per cluster
- **User input sorting**, to *ask information* in the correct order to end users (for instance, username before password), and also new types (lists, booleans, etc)
- **Improved naming conventions**, to allow importing resources with the same name in different vCenter locations

- **Numerous web interface enchancements**, like automatic estimation of VM cost through the Showback mechanism, better VM information display in *Cloud View*, image upload resume option, improved user and group management dialogs and many more.
- **Better audit trail**, now the history records of VMs includes the UID of the user that perfomed the action

Multiple bugfixes and documentation improvements have been included in this version. For instance, deleting SSH keys from cloud view, importing images with correct size, VM contextualization persistance across reboots, disk not removed if detached in poweroff state, context disk not displayed in Sunstone, skip import of resources if no permissions available, and a long list of other bugfixes and enhancements that can be consulted in the development portal.

vOneCloud 3.0 has been certified with *support for vSphere 5.5, 6.0 and 6.5*.

Warning: Upgrade to 3.0 from previous versions cannot be performed automatically. If you hold an active support subscription, please contact OpenNebula Systems to schedule a vOneCloud upgrade.

1.2 Upgrade

Upgrade to 3.0 from previous versions cannot be performed automatically. If you hold an active support subscription, please contact OpenNebula Systems to schedule a vOneCloud upgrade.

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.

1.3 System Requirements

Warning: It is advised to manage one vCenter by only one vOneCloud (ie, do not manage the same vCenter from two different vOneClouds). Otherwise VMs from both server will clash and produce errors.

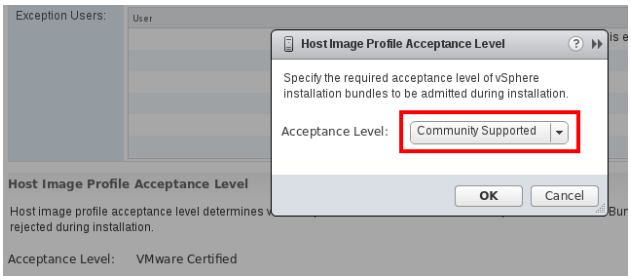
The following components are needed to be present in the infrastructure to implement a cloud infrastructure run by vOneCloud:

Component	Observations
vCenter 5.5/6.0/6.5	• 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 • Ideally, all ESX belonging to the same vCenter cluster to be exposed to vOneCloud need to share at least one datastore among them, although this is not a hard requirement. • VMs that will be instantiated through vOneCloud need to be saved as VMs Templates in vCenter. vOneCloud only creates new VMs by instantiating VM Templates.
ESX 5.5/6.0/6.5	• 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.
IE (>= 9), Firefox (> 3.5) and Chrome	Other browsers, including Safari, are **not** supported and may not work well. Note that IE11 is NOT supported with compatibility mode enabled.

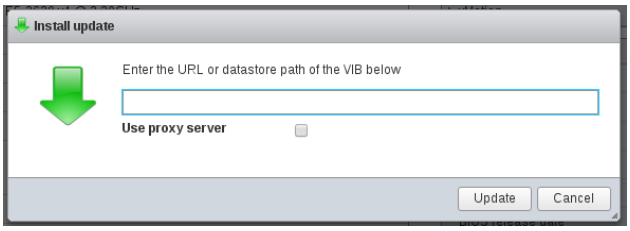
Note: To enable VNC functionality for vOneCloud, repeat the following procedure for each ESX. The following package adds the VNC ruleset (port range 5900-65535) and permits access to these ports.

- ZIP
- VIB
- Allow custom VIB package to be installed (in the vSphere client)
  - Login the vSphere client
  - Go to Home -> Inventories -> Hosts and Clusters
  - Select the ESX host and its tab **Manage** or **Configure** (depends on the vSphere version)

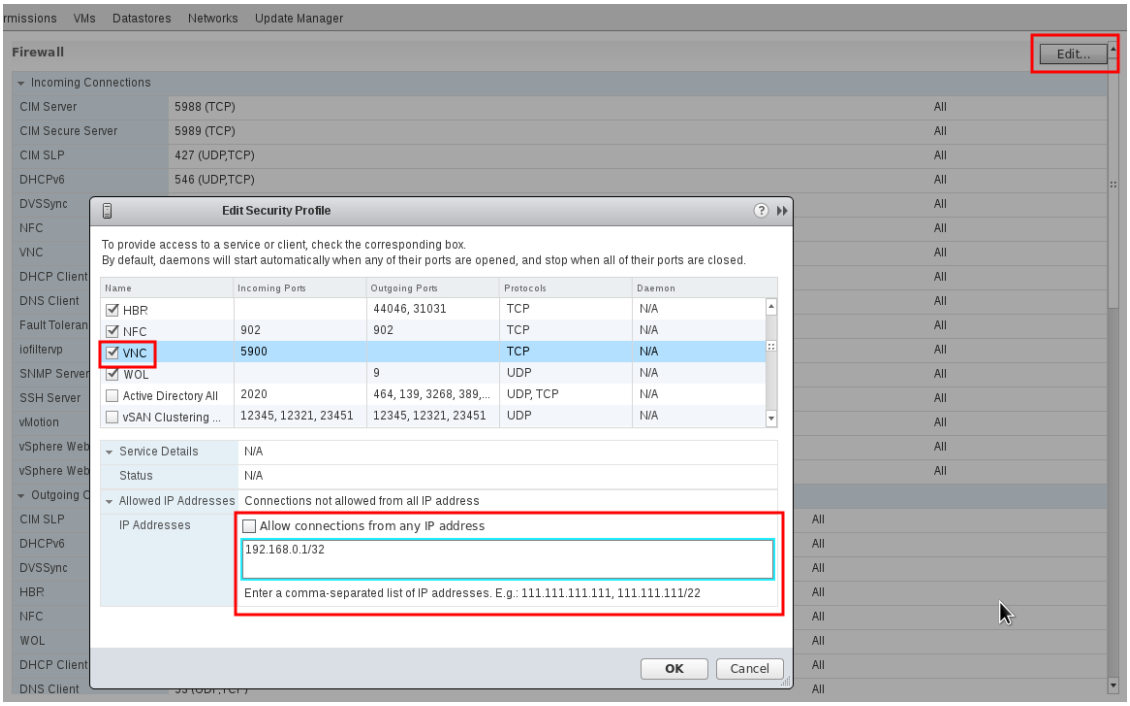
  - Select **Security Profile** in the **System category**
  - At the very bottom, select edit on **Host Image Profile Acceptance Level**
  - Switch to **Community Supported** and confirm with **OK**



- Install VIB package (in the ESX host UI)
  - Login the ESX host UI
  - Go to Help -> Update in top right corner
  - Provide the VIB URL or absolute local path and click on **Update**



- Restrict VNC access to the OpenNebula Front-end only (in the vSphere client)
  - Go back again to the ESX host details in the vSphere client
  - Reload the vSphere page to see current data
  - Check again **Security Profile** in the **System category**, look on the Firewall/Incoming Connections for new **VNC** item
  - Click on **Edit** for the Firewall
  - Find the VNC and optionally restrict access only to your OpenNebula Front-end (e.g. for 192.168.0.1):



Also, 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:

- Delete operation leaves a poweroff instance registered in vCenter.
- Wrong import of vCenter VM Templates with NICs in Distributed vSwitches or Distributed Ports
- Spaces in VMDK names and dirnames not supported

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: - Migrate VM to different ESX clusters
No spaces in VMDKs	VMDKs with spaces in their names or paths (ie, folders that contain them) are not supported for importing, attaching or uploading
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
Browser Adblock plug ins	Features like VNC and VM log viewer may be affected by Adblock plug ins. Please disable these plug ins if you are experiencing issues
Cloning imported VMs	Cloning in vCenter an imported VM will result in a VM that cannot be imported again. Please instantiate from templates and import the resulting VMs, instead of cloning already imported VMs.
VLAN ID not reported	vCenter precreated networks are not imported with the VLAN ID information, although they'll work as expected
Wrong capcity check at import time	When a vCenter template or wild VM is imported into OpenNebula, the virtual disks are imported, and vOneCloud tries to fit them in the DS. If not enough space are left, the import may fail.

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, and in some cases like for instance Storage DRS, vOneCloud offers functionality present only in VMware with the Enterprise Plus license.

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

2.2 What Is?

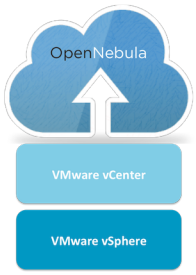
vOneCloud

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 management 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 3.0 includes:

CentOS	7.3
OpenNebula	5.4.0



The following table summarizes the benefits of vOneCloud:

Powerful	Virtual data centers, self-service, Virtual Routers on VMware environments
Cost Effective	Free, there are no license costs, all components 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 pre-configured 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 and powered off Virtual Machines
  - Import existing Datastores and VMDK images
  - Import existing Storage Pods
- **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 Confinement, restrict vOneCloud users/groups to a subset of hardware specified by a Resource Pool
- **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
  - Snapshot management
  - Contextualization capabilities, including the ability to run any script at VM boot time
  - VM capacity resizing (CPU and Memory)
  - Full networking support: vCenter Network and Distributed vSwitch import and creation
  - Full storage support: non persistent images and volatile disks
  - Connect Virtual Networks using a Virtual Router
  - Attach/detach network interfaces functionality
  - VNC connection to VMs, including the ability to set keymap
  - Attach/detach disk functionality
  - Save a running VM as a VM Template
  - Disk resize at boot time and in poweroff state
- **Virtualization Management**
  - Folder management
  - Limit and monitor VM network and disk consumption
  - Linked clone support
  - Import VM Templates with network and storage information
  - Chose datastore and Resource Pool where VMs will be deployed to
  - Instantiate to persistent to easily crate a VM Template catalog

- **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
  - Powerful user, group and role management
  - Showback functionality to report resource usage cost
- **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 Management Platform will be up to the task

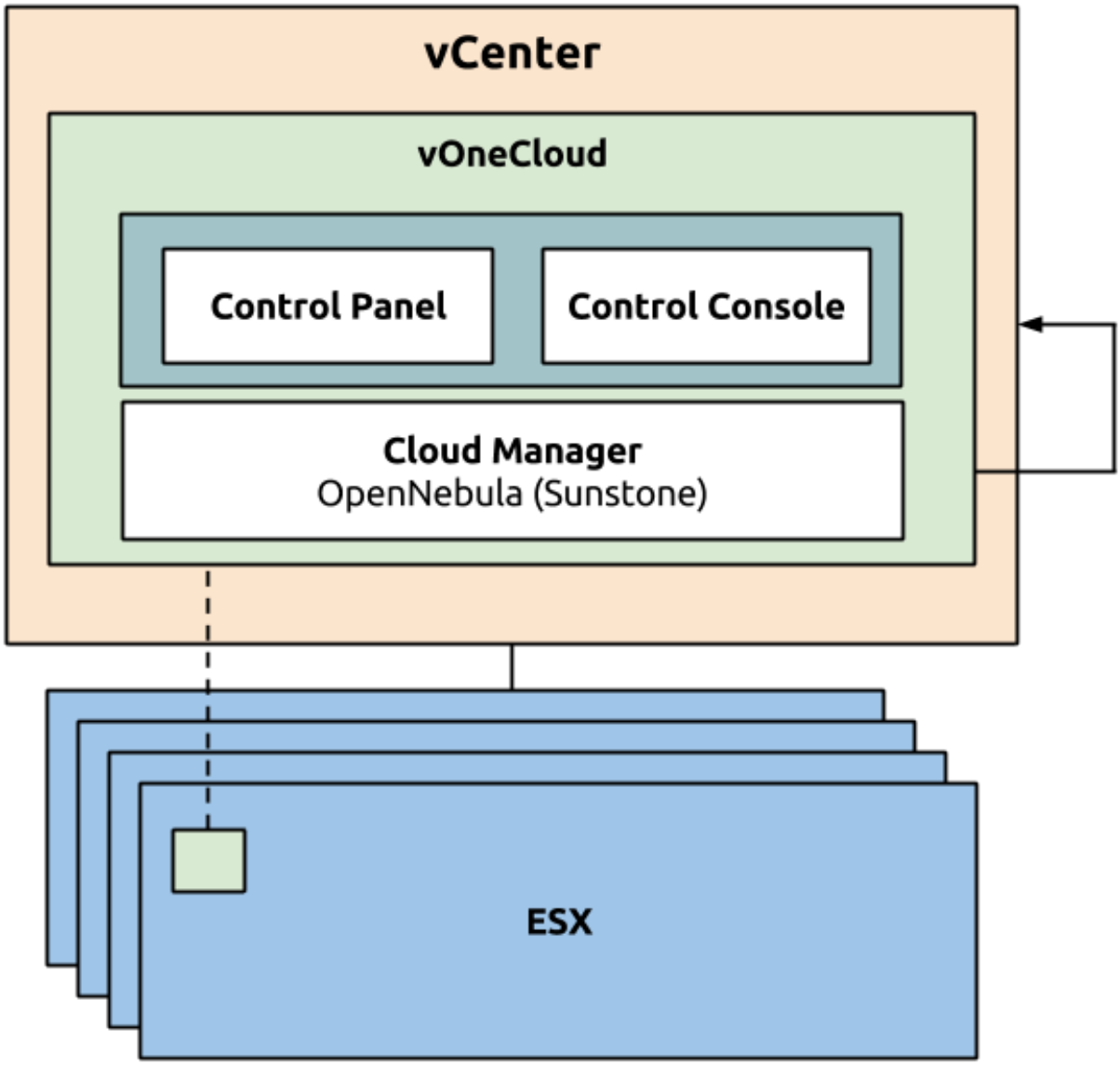
vOneCloud additionally brings new configuration and upgrade tools:

- **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. If you are building a large-scale cloud, are interested in the federation of multiple controller instances, or want to integrate with third party components, customize the product or manage open source hypervisors, we recommend an installation of OpenNebula.

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 on 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 accessible 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 (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 access is disabled by default in the appliance. You can log in either opening an alternate TTY on a Virtual Machine Console in vSphere's client or enabling SSH using vOneCloud's Control Panel.

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 pre configured 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

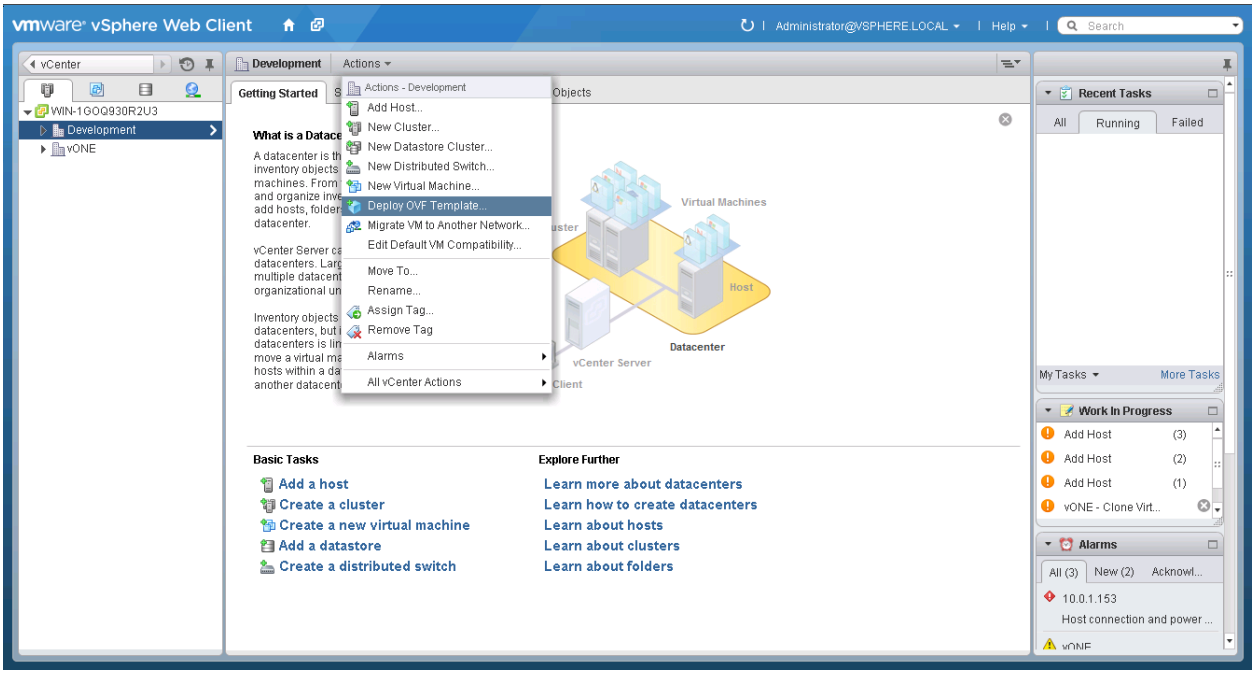
You can import this OVA appliance to your vCenter infrastructure. It is based on CentOS 7 and has the VMware tools enabled.

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*.

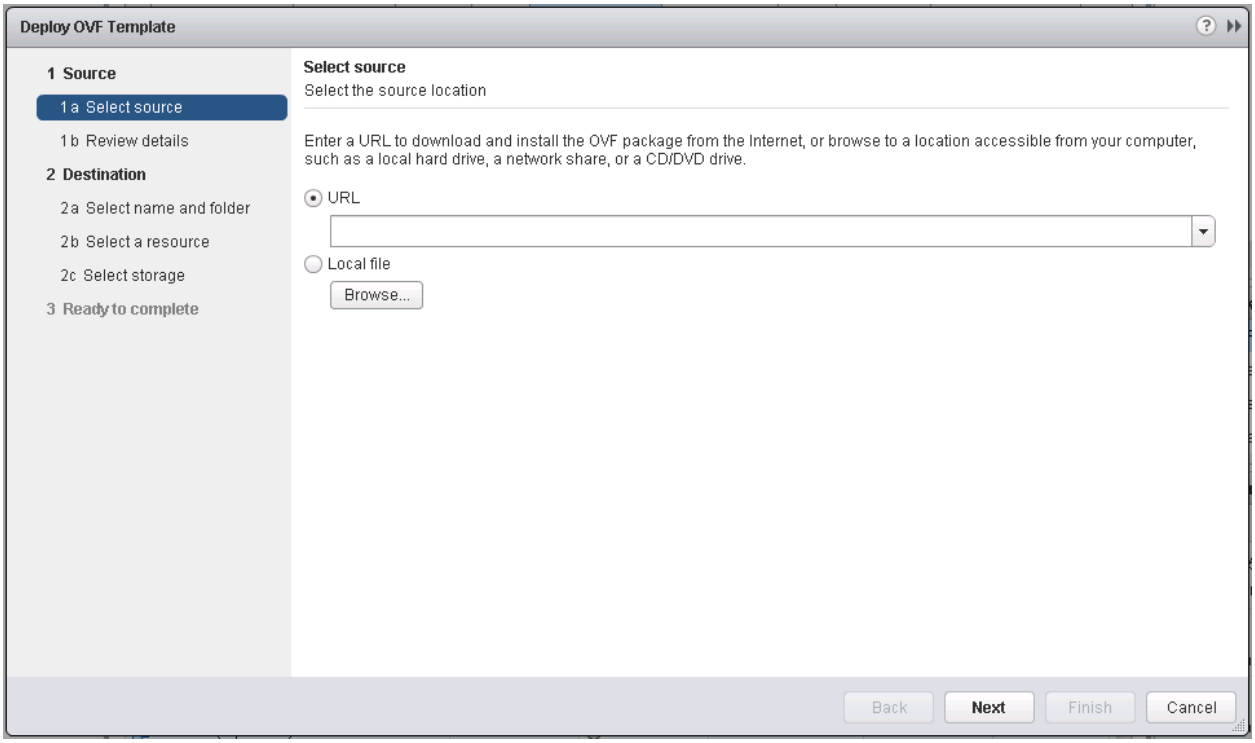
Follow the next steps to deploy a fully functional vOneCloud:

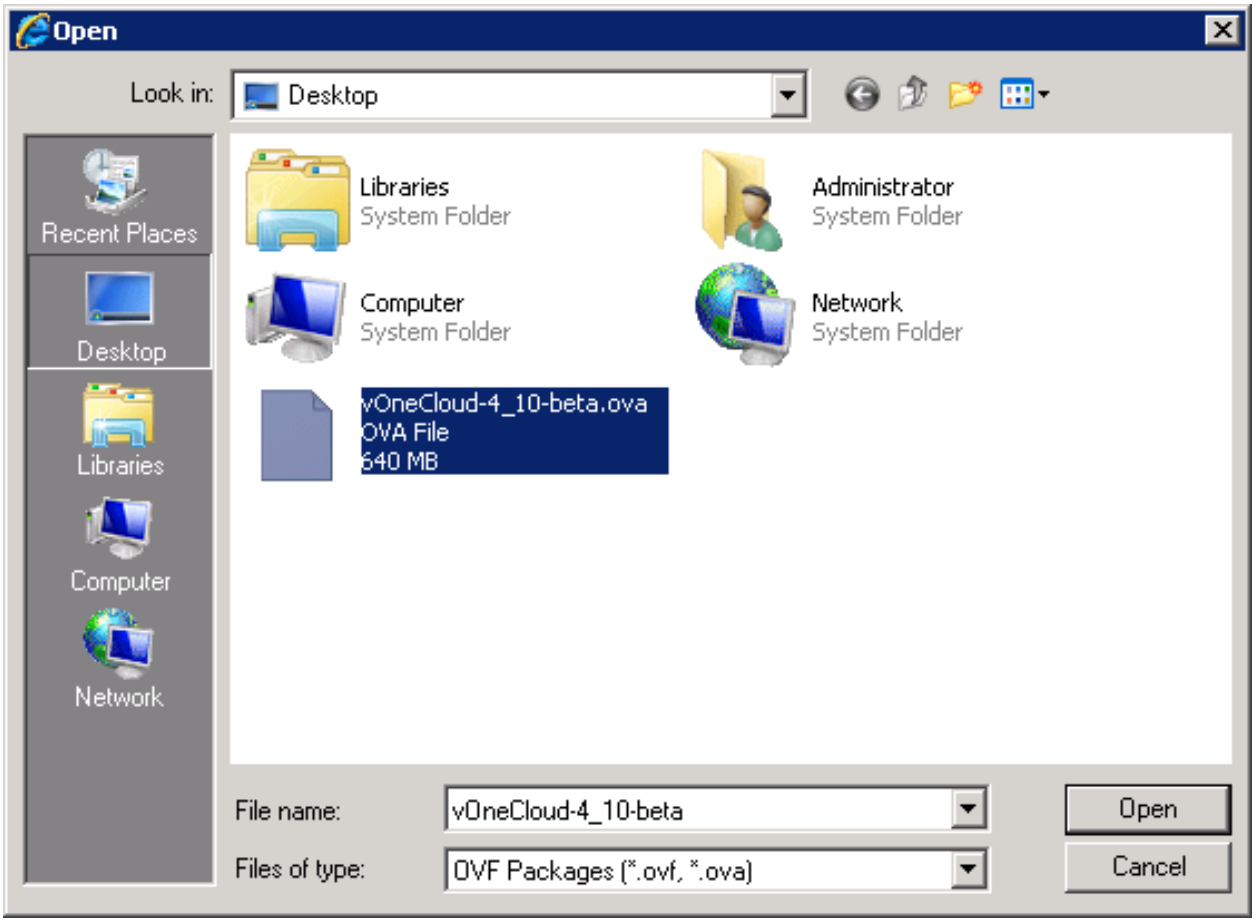
3.2.1 Step 1. Deploying the OVA

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

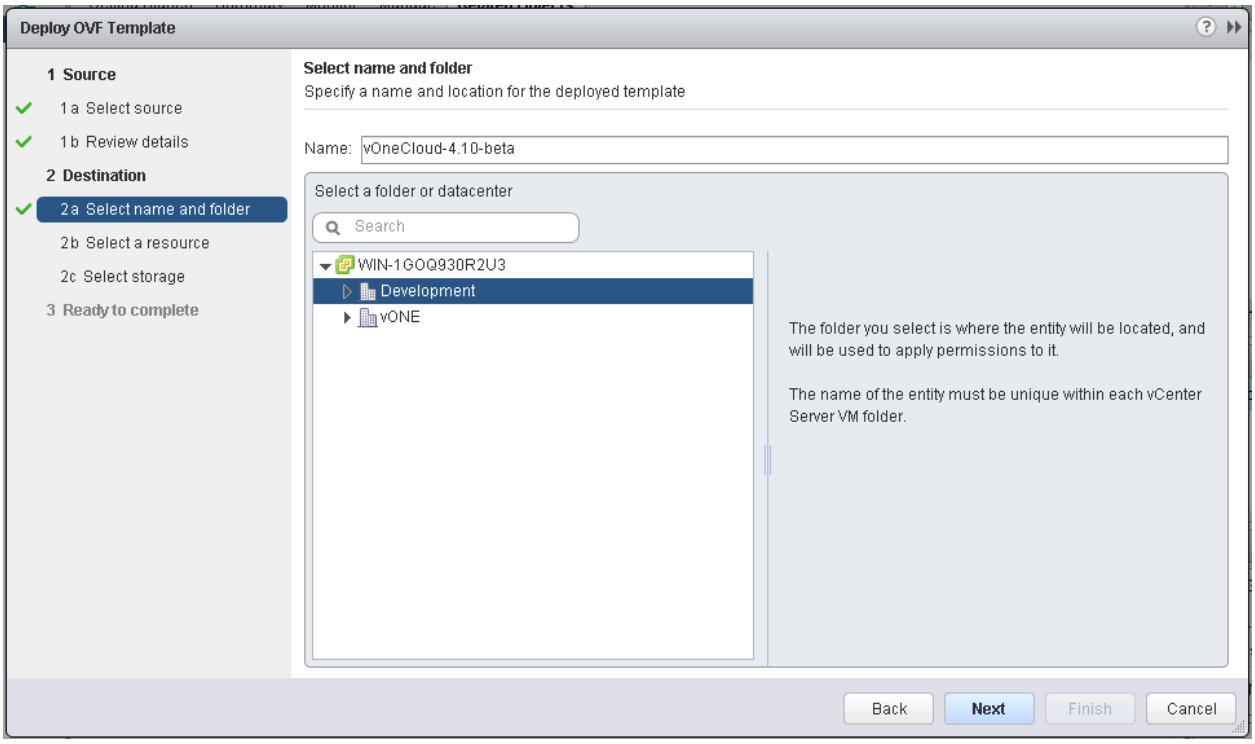


You have the option now to input the URL of the appliance (you can find it at the top of this page), or if you have previously downloaded it, you can simply browse to the download path as such:

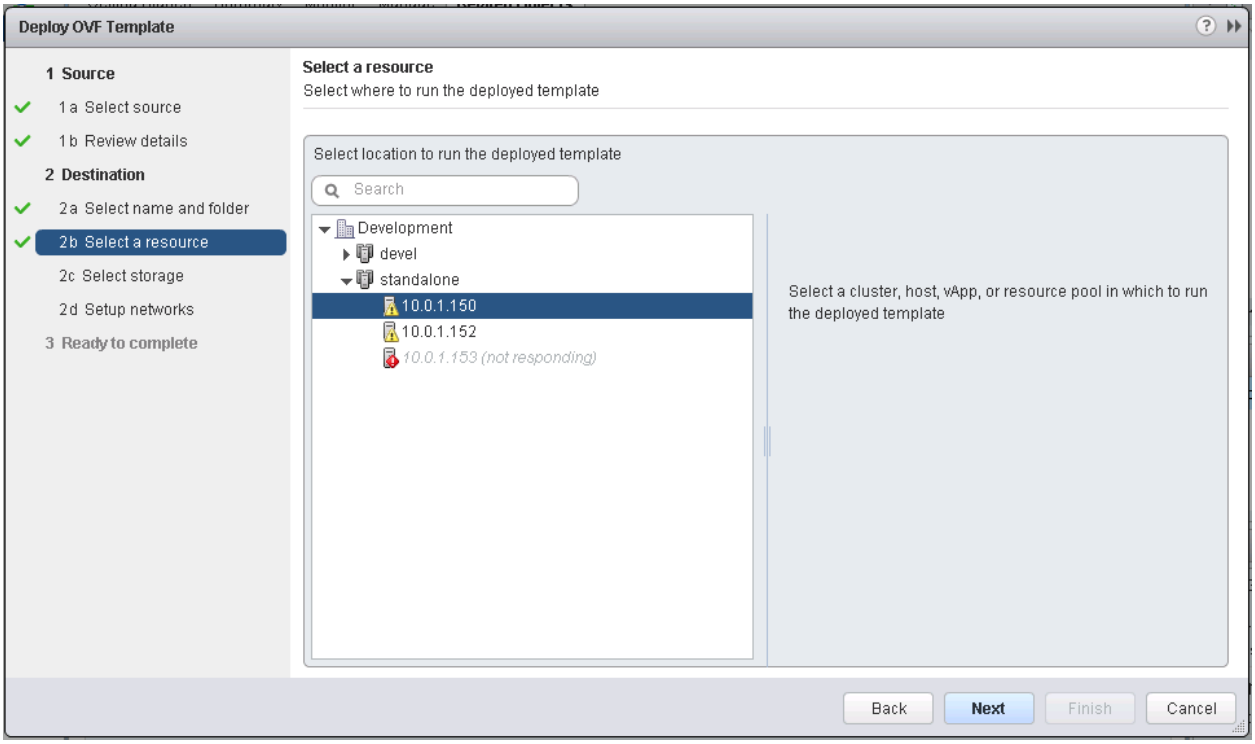




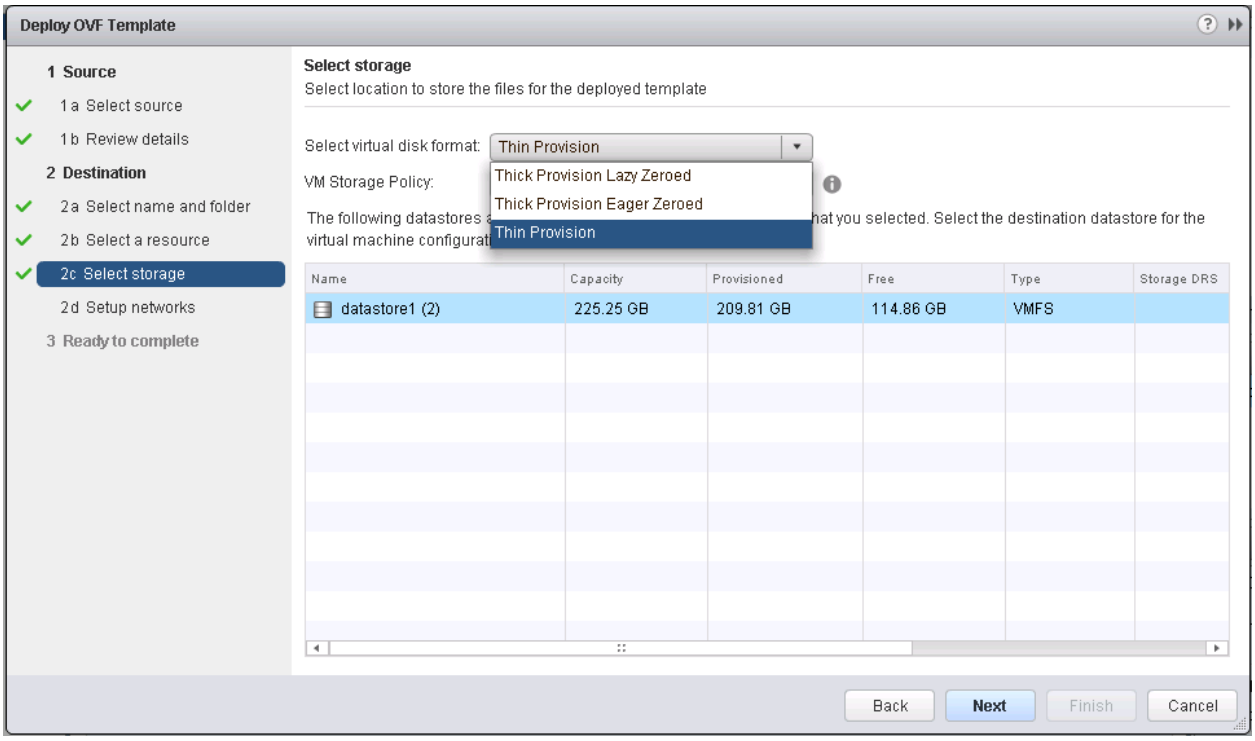
Select the name and folder:



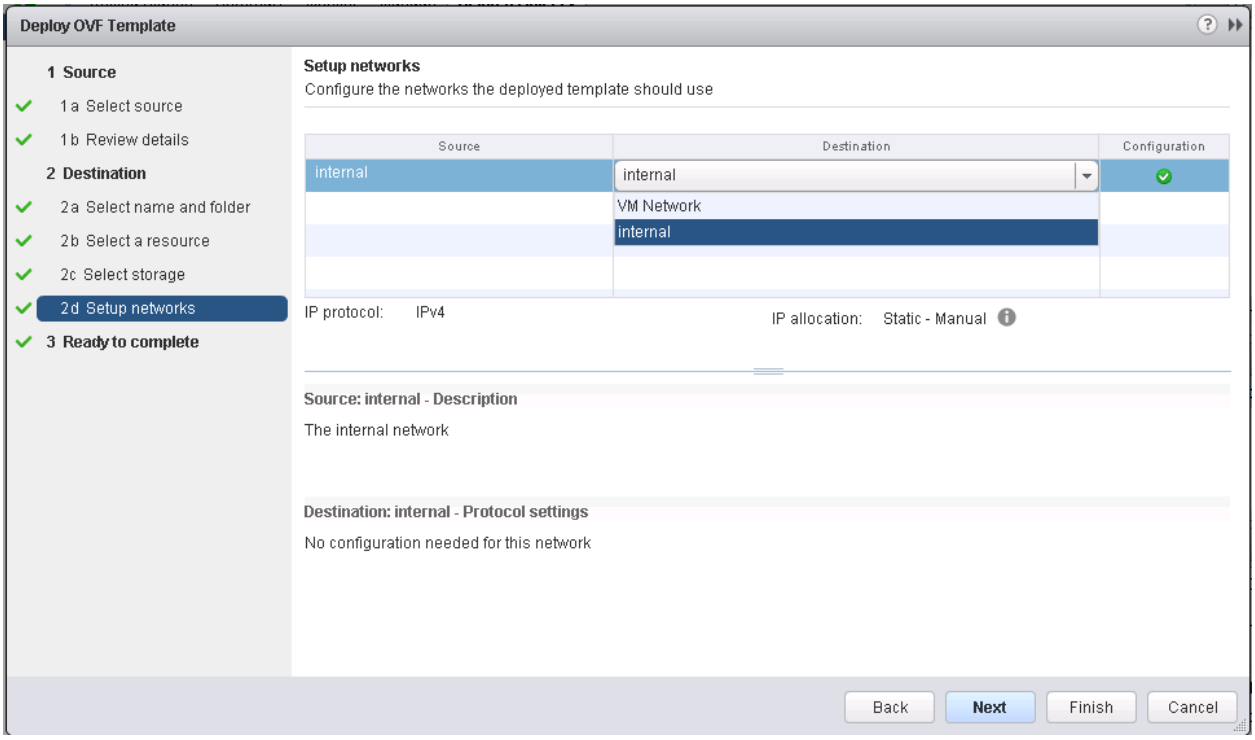
Select a resource to run the appliance:



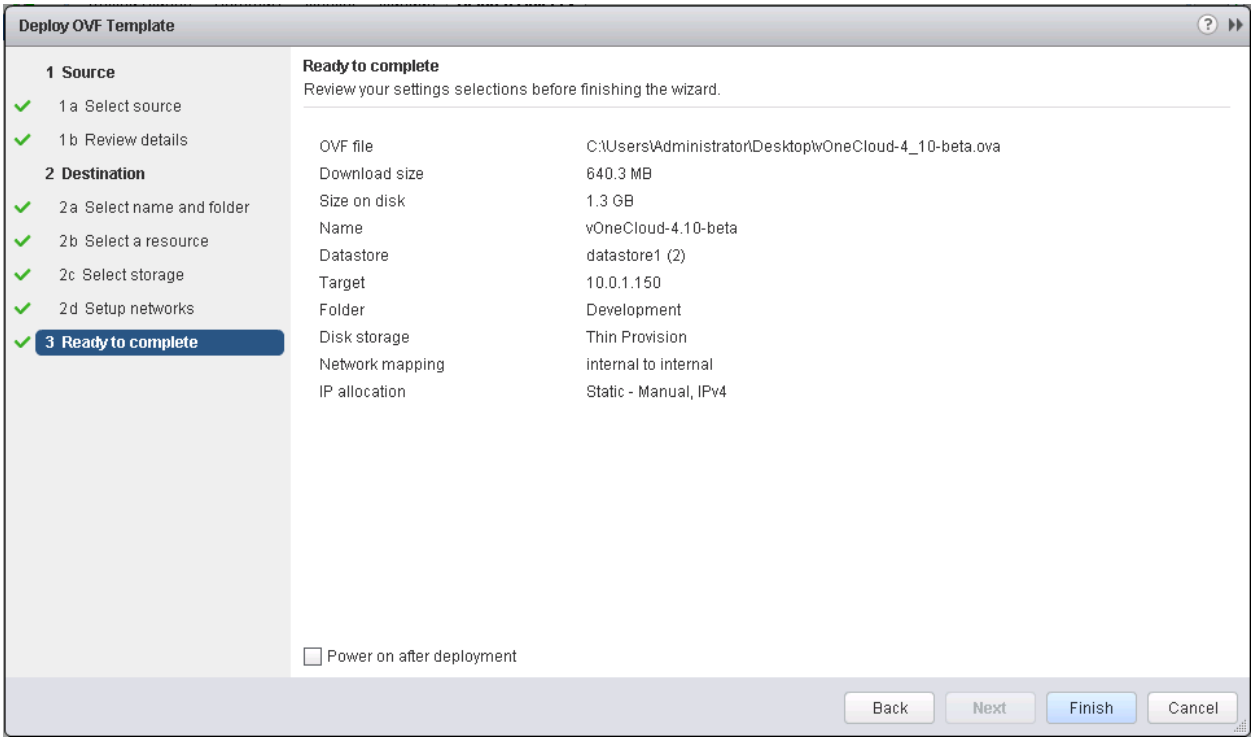
Select the datastore:



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

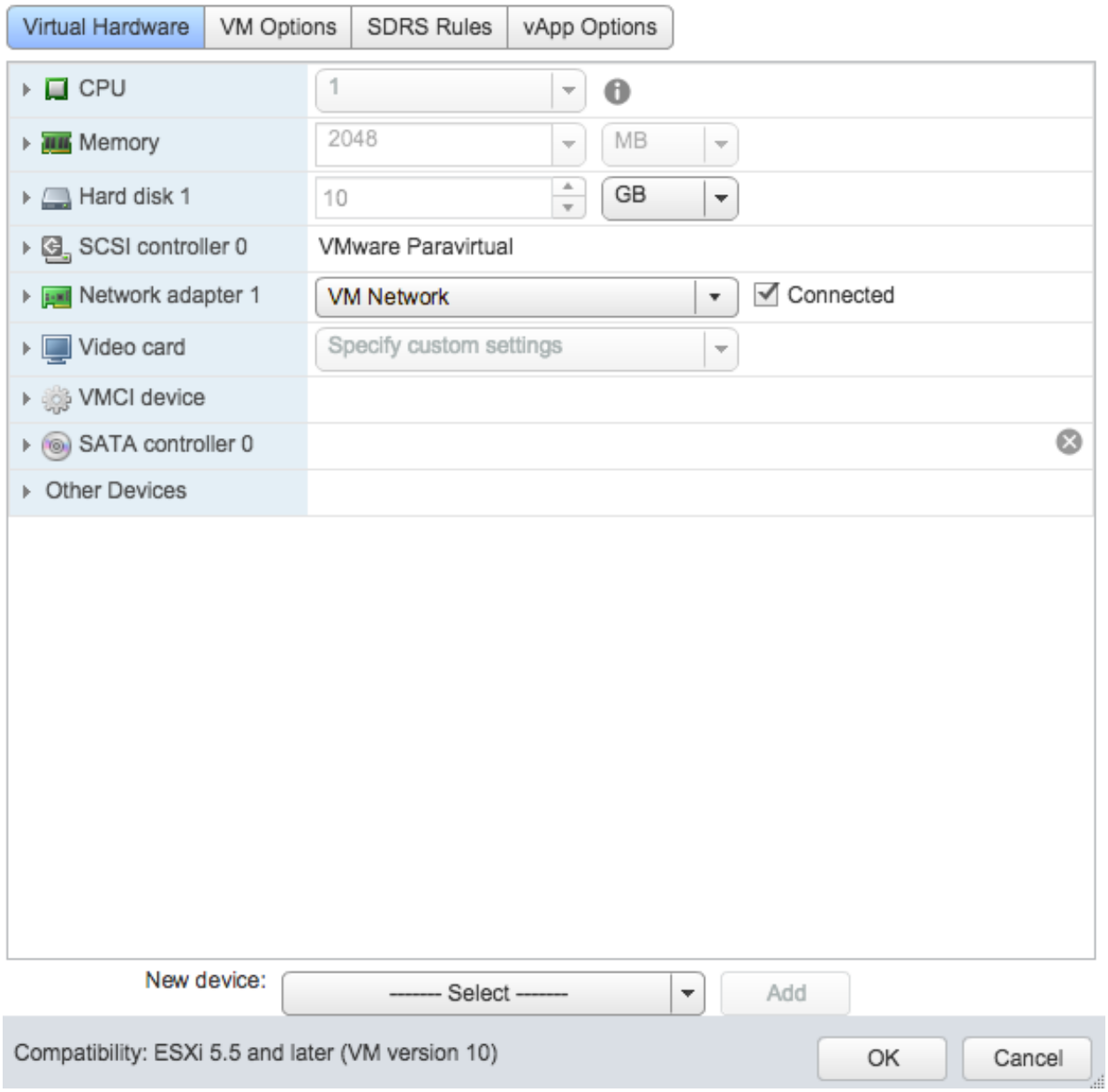


Review the settings selection and click finish. Wait for the Virtual Machine to appear in the cluster.

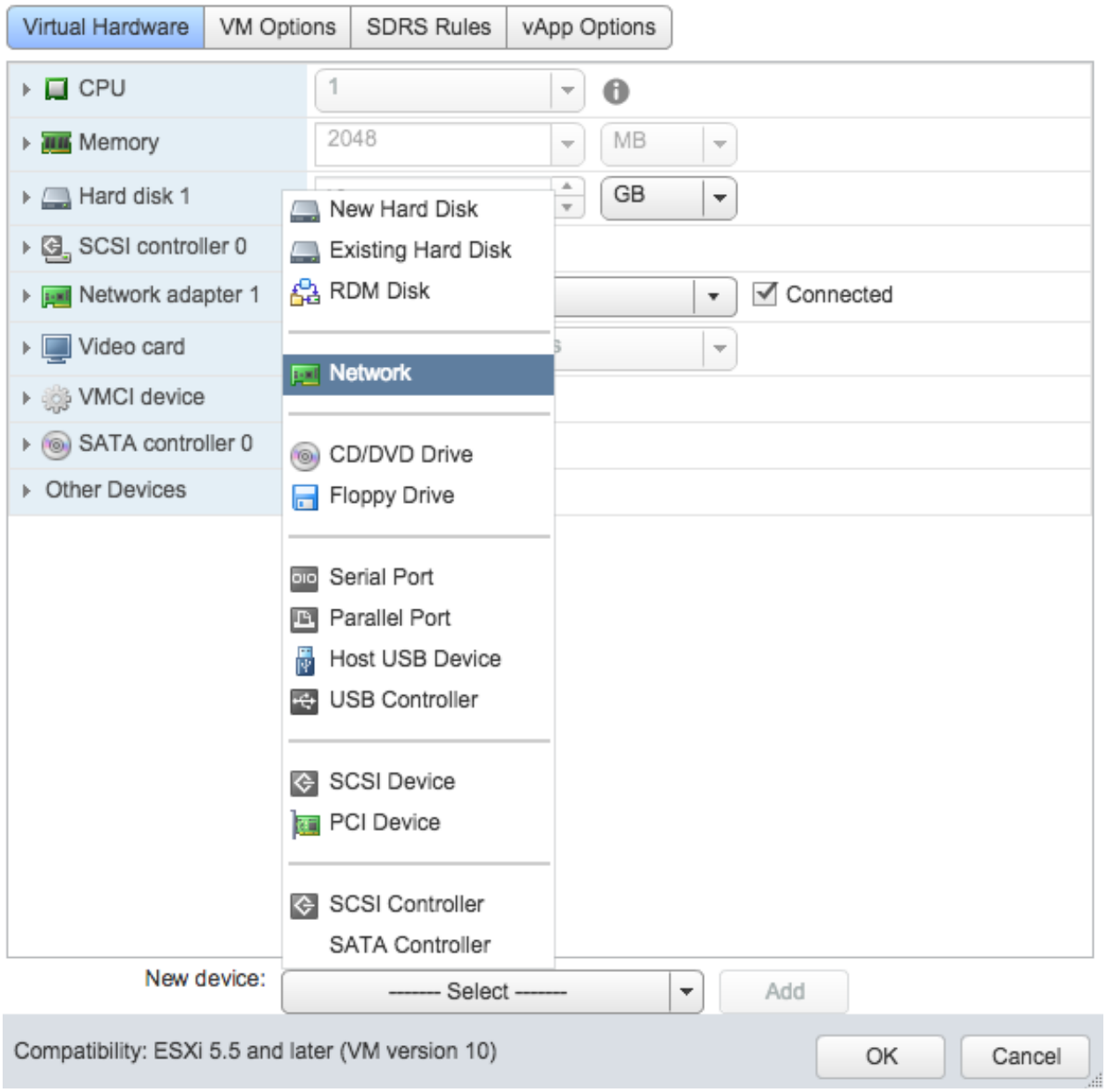


After importing the vOneCloud OVA, and before powering it on, the vOneCloud Virtual Machine can be edited to, for instance, add a new network interface, increase the amount of RAM, the available CPUs for performance, etc.

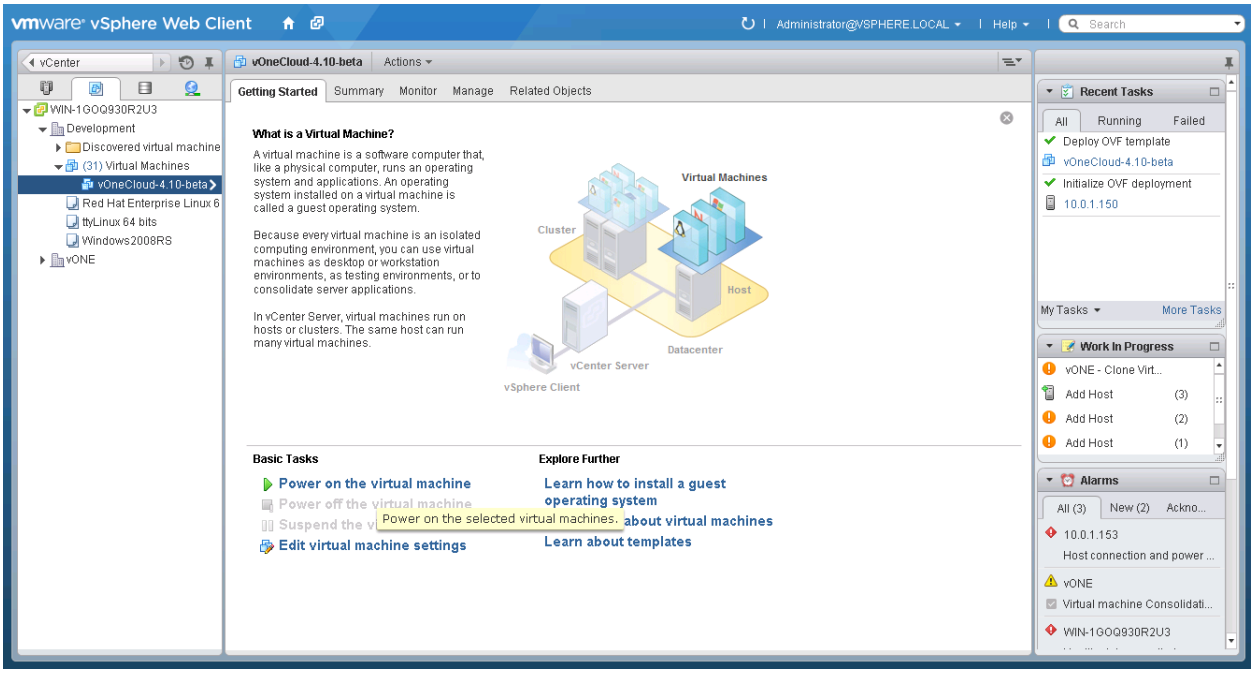
In order to achieve this, please right click on the vOneCloud VM, and select Edit Settings. The next dialog should pop up:



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):



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 VM console in vCenter the *vOneCloud Control Console*, showing this wizard:

If you are presented instead with the following:

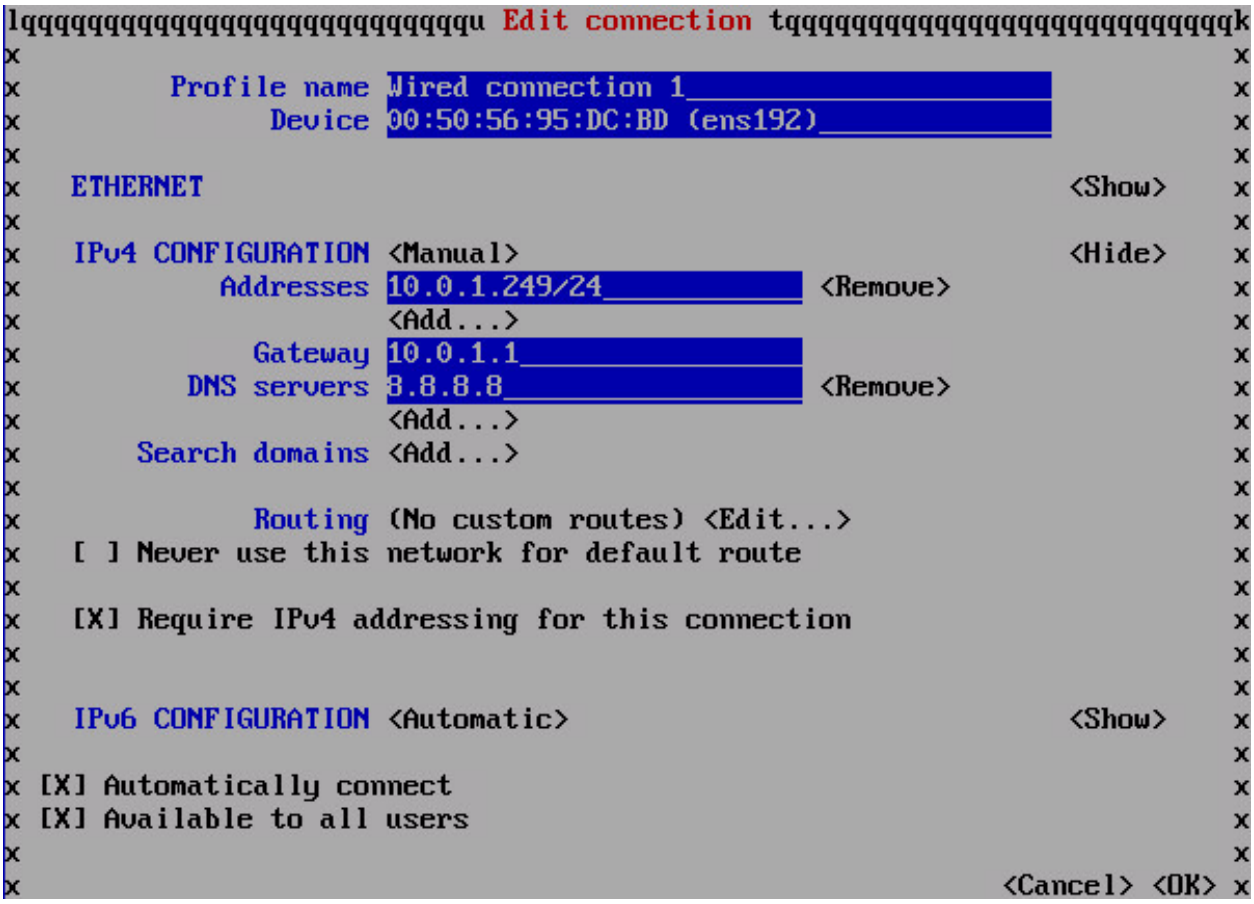
You are being presented with the wrong tty. You will need to press CTRL+ALT+F1 to access the Control Console.

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 "Wired 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.

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.

3.2.3 Step 3. vOneCloud Control Panel - Manage Services

The wizard points out that you can open the vOneCloud Control Panel by visiting *http://<appliance_ip>:8000* in your browser and using the *oneadmin* account and password just chosen.

> **Warning:** Are you using a self-signed certificate? If that's the case you must accept it too in port 29876. In order to do so open https://...:29876 and verify that there is no SSL related warning. Click here for more information.

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

- Check for new vOneCloud versions and manage automatic upgrades.
- Configure Active Directory / LDAP integration
- Configure SSL (https) access
- Enable/Disable ssh
- Start the OpenNebula services

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.

3.3 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 cluster as well as any already defined VM Template and Networks.

You will need the IP or hostname of the vCenter server, as well as a user declared as Administrator in vCenter.

Alternatively, in some enterprise environments declaring the user as Administrator is not allowed, in that case, you will need to grant the following permissions to a user depending on what OpenNebula's functionality you want to enable:

Privileges	Notes
VirtualMachine.Interact.DeviceConnection	Required by a virtual machine reconfigure action
VirtualMachine.Interact.SetCDMedia	Required by a virtual machine reconfigure action
VirtualMachine.Interact.SetFloppyMedia	Required by a virtual machine reconfigure action
VirtualMachine.Config.Rename	Required by a virtual machine reconfigure action
VirtualMachine.Config.Annotation	Required by a virtual machine reconfigure action
VirtualMachine.Config.AddExistingDisk	Required by a virtual machine reconfigure action
VirtualMachine.Config.AddNewDisk	Required by a virtual machine reconfigure action
VirtualMachine.Config.RemoveDisk	Required by a virtual machine reconfigure action
VirtualMachine.Config.CPUCount	Required by a virtual machine reconfigure action
VirtualMachine.Config.Memory	Required by a virtual machine reconfigure action
VirtualMachine.Config.RawDevice	Required by a virtual machine reconfigure action
VirtualMachine.Config.AddRemoveDevice	Required by a virtual machine reconfigure action
VirtualMachine.Config.Settings	Required by a virtual machine reconfigure action
VirtualMachine.Config.AdvancedConfig	Required by a virtual machine reconfigure action
VirtualMachine.Config.SwapPlacement	Required by a virtual machine reconfigure action
VirtualMachine.Config.HostUSBDevice	Required by a virtual machine reconfigure action
VirtualMachine.Config.DiskExtend	Required by a virtual machine reconfigure action
VirtualMachine.Config.ChangeTracking	Required by a virtual machine reconfigure action
VirtualMachine.Provisioning.ReadCustSpecs	Required by a virtual machine reconfigure action
VirtualMachine.Inventory.CreateFromExisting	Required by a virtual machine reconfigure action
VirtualMachine.Inventory.CreateNew	Required by a virtual machine reconfigure action
VirtualMachine.Inventory.Move	Required by a virtual machine reconfigure action
VirtualMachine.Inventory.Register	Required by a virtual machine reconfigure action
VirtualMachine.Inventory.Remove	Required by a virtual machine reconfigure action
VirtualMachine.Inventory.Unregister	Required by a virtual machine reconfigure action
VirtualMachine.Inventory.Delete	Required to delete a virtual machine
VirtualMachine.Provisioning.DeployTemplate	Required to deploy a virtual machine from a particular template
VirtualMachine.Provisioning.CloneTemplate	Required to create a copy of a particular template
VirtualMachine.Interact.PowerOn	Required to power on a virtual machine
VirtualMachine.Interact.PowerOff	Required to power off or shutdown a virtual machine
VirtualMachine.Interact.Suspend	Required to suspend a virtual machine
VirtualMachine.Interact.Reset	Required to reset/reboot a VM's guest Operating System

Table 3.1 – continued from previous page

VirtualMachine.Inventory.Delete	Required to delete a virtual machine or template
VirtualMachine.State.CreateSnapshot	Required to create a new snapshot of a virtual machine.
VirtualMachine.State.RemoveSnapshot	Required to remove snapshots from a virtual machine
VirtualMachine.State.RevertToSnapshot	Required to revert a virtual machine to a particular snapshot
Resource.AssignVirtualMachineToResourcePool	Required to assign a resource pool to a virtual machine
Resource.ApplyRecommendation	On all Storage Pods (Storage DRS cluster) represented by OpenNebula
Datastore.AllocateSpace	On all VMFS datastores represented by OpenNebula
Datastore.LowLevelFileOperations	On all VMFS datastores represented by OpenNebula
Datastore.RemoveFile	On all VMFS datastores represented by OpenNebula
Datastore.Browse	On all VMFS datastores represented by OpenNebula
Datastore.FileManagement	On all VMFS datastores represented by OpenNebula
Network.Assign	Required on any network the Virtual Machine will be connected to
System.Read	Required to rename Uplink port group for a distributed switch only if you want O
Host.Config.Network	Required an all **ESX hosts** where you want OpenNebula to create, update or delet
DVSwitch.CanUse	Required to connect a VirtualEthernetAdapter to a distributed virtual switch either
DVSwitch.Create	Required if you want OpenNebula to create distributed virtual switches
DVSwitch.HostOp	Required if you want OpenNebula to create distributed virtual switches
DVSwitch.PortSetting	Required if you want OpenNebula to create distributed virtual switches
DVSwitch.Modify	Required if you want OpenNebula to create distributed virtual switches
DVSwitch.Delete	Required if you want OpenNebula to destroy a distributed virtual switches that wa
DVPortgroup.Create	Required if you want OpenNebula to create distributed port groups
DVPortgroup.CanUse	Required to connect a VirtualEthernetAdapter to a distributed virtual port group e
DVSwitch.Modify	Required if you want OpenNebula to create distributed port groups
DVPortgroup.Delete	Required if you want OpenNebula to destroy a distributed port group that was pre

Note: For security reasons, you may define different users to access different ESX Clusters. A different user can be defined in OpenNebula per ESX cluster, which is encapsulated in OpenNebula as an OpenNebula host.

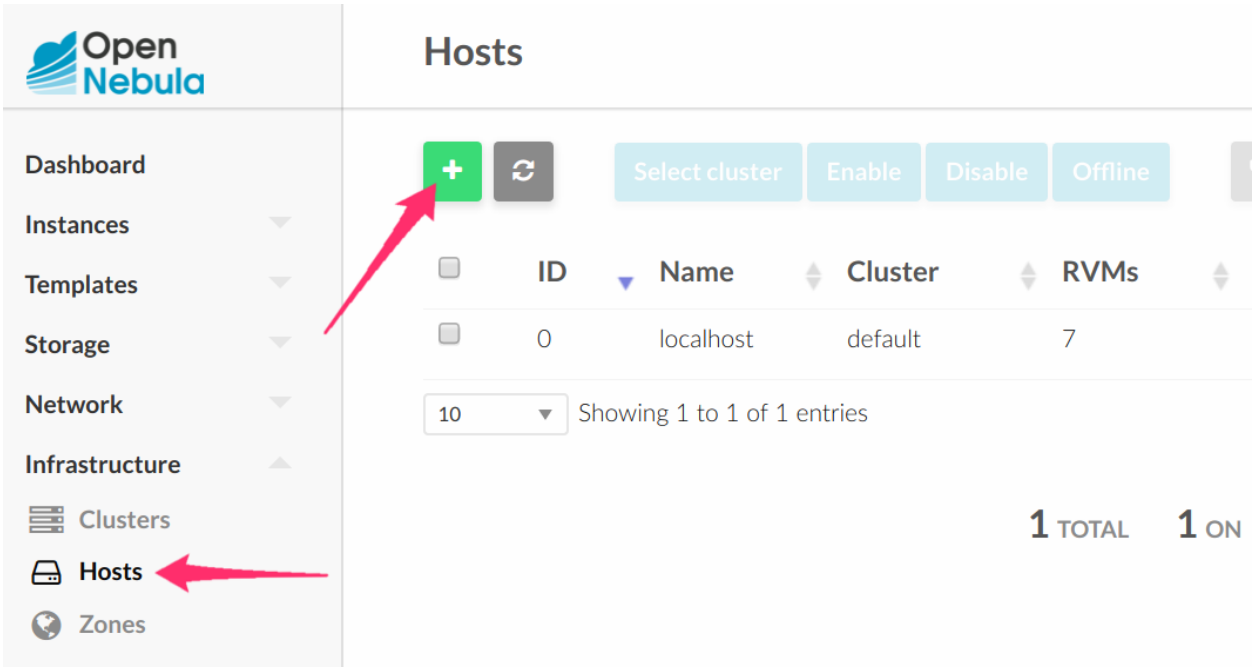
3.3.1 Step 1. Sunstone login

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

The *CloudAdmin* user comes pre configured and is the **Cloud Administrator**, in full control of all the physical and virtual resources and using the vCenter view. Views will be explained later in its own section.

3.3.2 Step 2. Acquire vCenter Resources

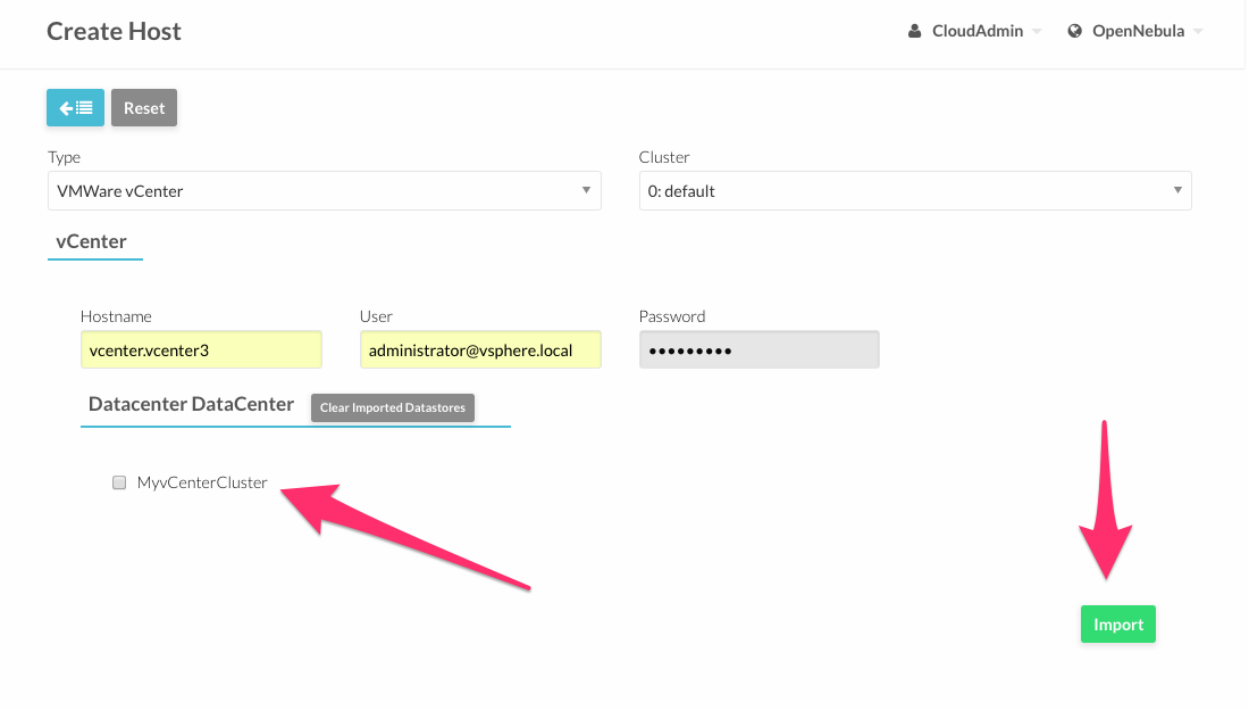
To import new vCenter clusters to be managed in vOneCloud, proceed in Sunstone to the `Infrastructure --> Hosts` tab and click on the "+" green icon.



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

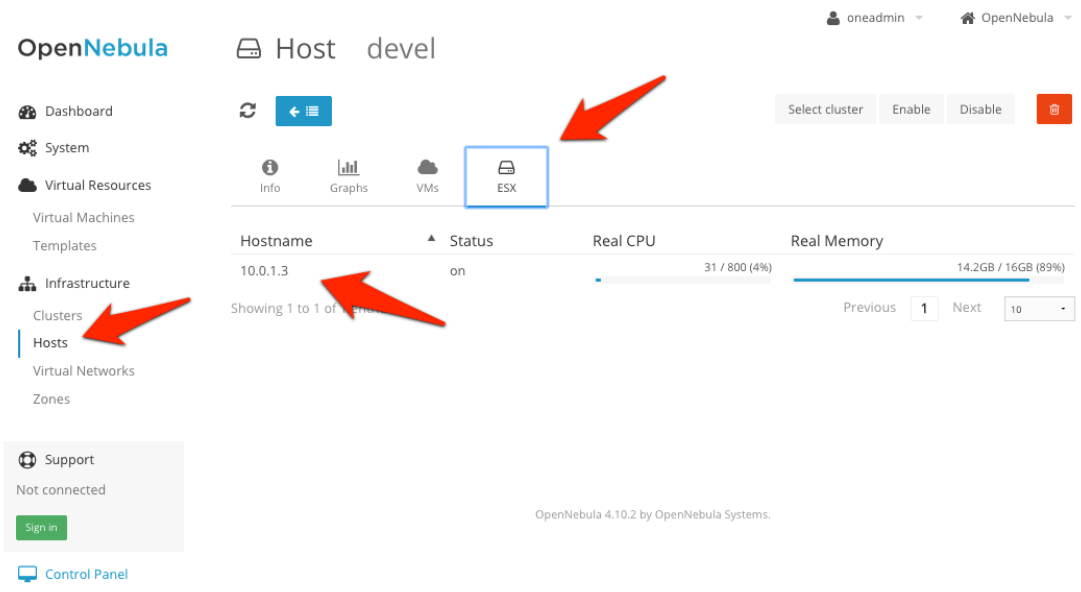
In the dialog that pops up, select vCenter as Type in the drop-down. You now need to fill in the data according to the following table:

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



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

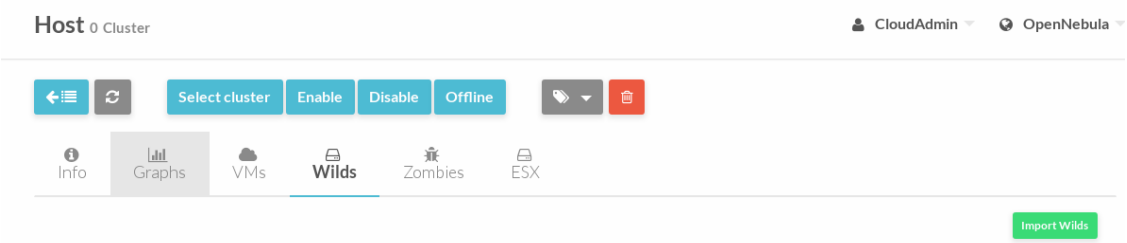
Note: Take into account that one vCenter cluster (with all its ESX hosts) will be represented as one vOneCloud host.



3.3.3 Step 3. Import / Reacquire vCenter Resources

Existing VMs

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 life-cycle, 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 with some exceptions. The following operations *cannot* be performed on an imported VM:

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

Once a Wild VM is imported, vOneCloud will reconfigure the vCenter VM so VNC connections can be established once the VM is monitored. **Datastores and Images**

Datastores and VMDK images can be imported / reacquired from the `Storage --> Datastores` and `Storage --> Images` respectively. Since datastores are going to be used to hold the images from VM Templates, all datastore **must** be imported before VM Template import.

vCenter datastores hosts VMDK files and other file types so VMs and templates can use them, and these datastores can be represented in OpenNebula as both an Images datastore and a System datastore:

- Images Datastore. Stores the images repository. VMDK files are represented as OpenNebula images stored in this datastore.
- System Datastore. Holds disk for running virtual machines, copied or cloned from the Images Datastore.

For example, if we have a vcenter datastore called ”nfs”, when we import the vCenter datastore into OpenNebula, two OpenNebula datastores will be created as an Images datastore and as a System datastore pointing to the same vCenter datastore.

Note: If the vCenter instance features a read only datastore, please be aware that you should disable the SYSTEM representation of the datastore after importing it to avoid OpenNebula trying to deploy VMs in it.

When an image or a datastore is imported, vOneCloud will generate a name automatically that prevents conflicts if you try to import several files with the same name but that are located in different folders inside the datastore, or try to import datastores with the same name in different vCenter instances. The image name contains the file’s name, the datastore’s name and a 12 character hash, whereas the datastore contains the datastore name, the vcenter instance name, the datacenter where it lives and the datastore type between parentheses. These names can be changed once the image or datastore has been imported.

When the vCenter hypervisor is used we have three OpenNebula image types:

- OS: A bootable disk Image. Every VM template must define one DISK referring to an Image of this type. These images can be imported or uploaded.

- CDROM: These Images are read-only data. These images can also be imported or uploaded.
- DATABLOCK: A datablock Image is a storage for data. These Images can be created from previous existing data (e.g uploading a VMDK file), or as an empty drive.

OpenNebula images can be also classified in persistent and non-persistent images:

- Non-persistent images. These images are used by at least one VM. It can still be used by other VMs. When a new VM using a non-persistent image is deployed a copy of the VMDK file is created.
- Persistent images. A persistent image can be use only by a VM. It cannot be used by new VMs. The original file is used, no copies are created.

Disks attached to a VM will be backed by a non-persistent or persistent image although volatile disks are also supported. Volatile disks are created on-the-fly on the target hosts and they are disposed when the VM is shutdown.

Datastore will be monitored for free space and availability. Images can be used for:

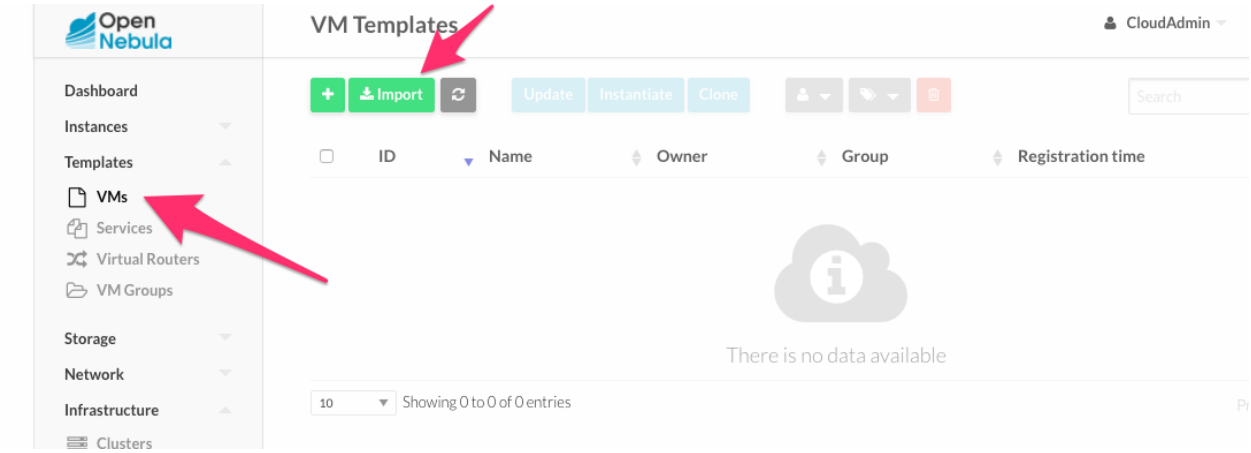
- disk attach/detach on VMs
- enrich VM Templates to add additional disks or CDROMs

VM Templates

> **Warning:** Since datastores are going to be used to hold the images from VM Templates, all datastore **must** be imported before VM Template import.

In vOneCloud, Virtual Machines are deployed from VMware VM Templates that must exist previously in vCenter and must be imported into vOneCloud. There is a one-to-one relationship between each VMware VM Template and the equivalent vOneCloud VM Template. Users will then instantiate the OpenNebula VM Template and OpenNebula will create a Virtual Machine clone from the vCenter template.

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.



When a VMware VM Template is imported, vOneCloud will detect any virtual disk and network interface within the template. For each virtual disk, vOneCloud will create an vOneCloud image representing each disk discovered in the template. In the same way, vOneCloud will create a network representation for each standard or distributed port group associated to virtual network interfaces found in the template. The imported 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 or removed to give VMs access to different networks
- Disks can be added or removed
- Capacity (MEMORY and CPU) can be modified
- VNC capabilities can be disabled

Check the *advanced features guide* for additional features available for VM Templates.

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. This value can be changed using a special attribute set in the vCenter cluster representation in vOneCloud, ie, the vOneCloud host. This attribute is called "VM_PREFIX", and will evaluate one variable, $i, to the id of the VM. A value of "one-$i-" in that parameter would have the same behaviour as the default. This attribute can be set in the "Attributes" section of the vOneCloud host, in the info panel that shows after clicking on the desire host.

Note: After a VM Template is cloned and booted into a vCenter Cluster it can access VMware advanced features and it can be managed through the OpenNebula provisioning portal -to control the life-cycle, add/remove NICs, make snapshots- or through vCenter (e.g. to move the VM to another datastore or migrate it to another ESX). OpenNebula will poll vCenter to detect these changes and update its internal representation accordingly.

Note: The name assigned to the template in OpenNebula contains the template's name, vCenter cluster's name and a 12 character hash. That name is used to prevent conflicts when several templates with the same name are found in a vCenter instance. Once the vCenter template has been imported, that OpenNebula's name can be changed to a more human-friendly name.

Networks

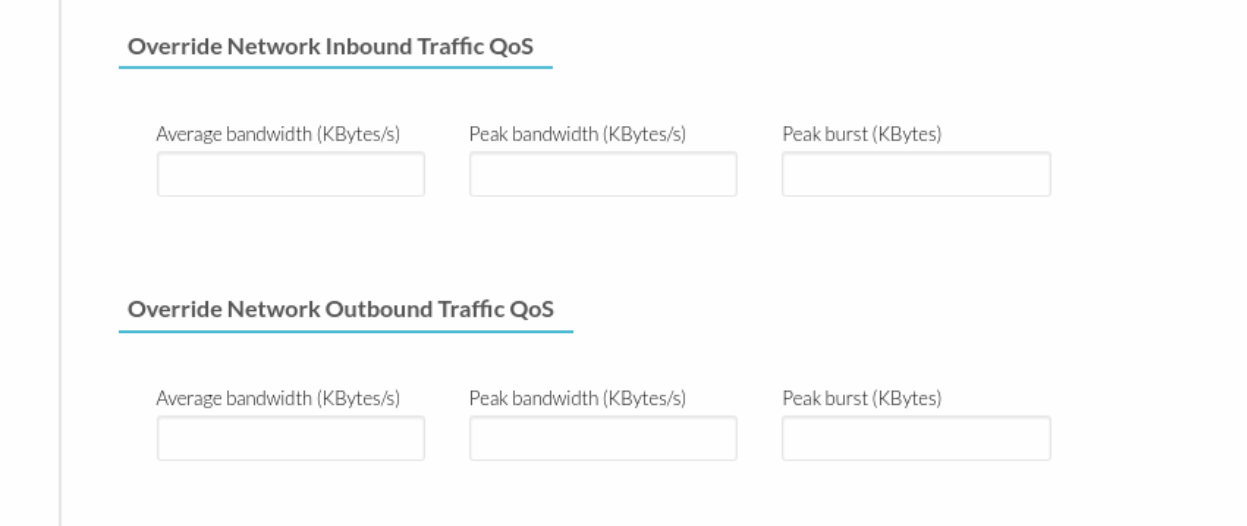
Similarly, **Networks** and Distributed vSwitches can also be imported / reacquired from using a similar `Import` button in `Infrastructure --> Virtual Networks`.

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 address, 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.

It is possible to limit the bandwidth of any VM NIC associated to a particular virtual network by using the Inbound/Outbound Traffic QoS values as seen in the next image.



Note: vOneCloud does not support spaces in VMDKs paths nor names.

Note: 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.

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

3.3.4 Step 4. 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 3** and click on Instantiate. Now you will be able to control the life cycle of the VM.

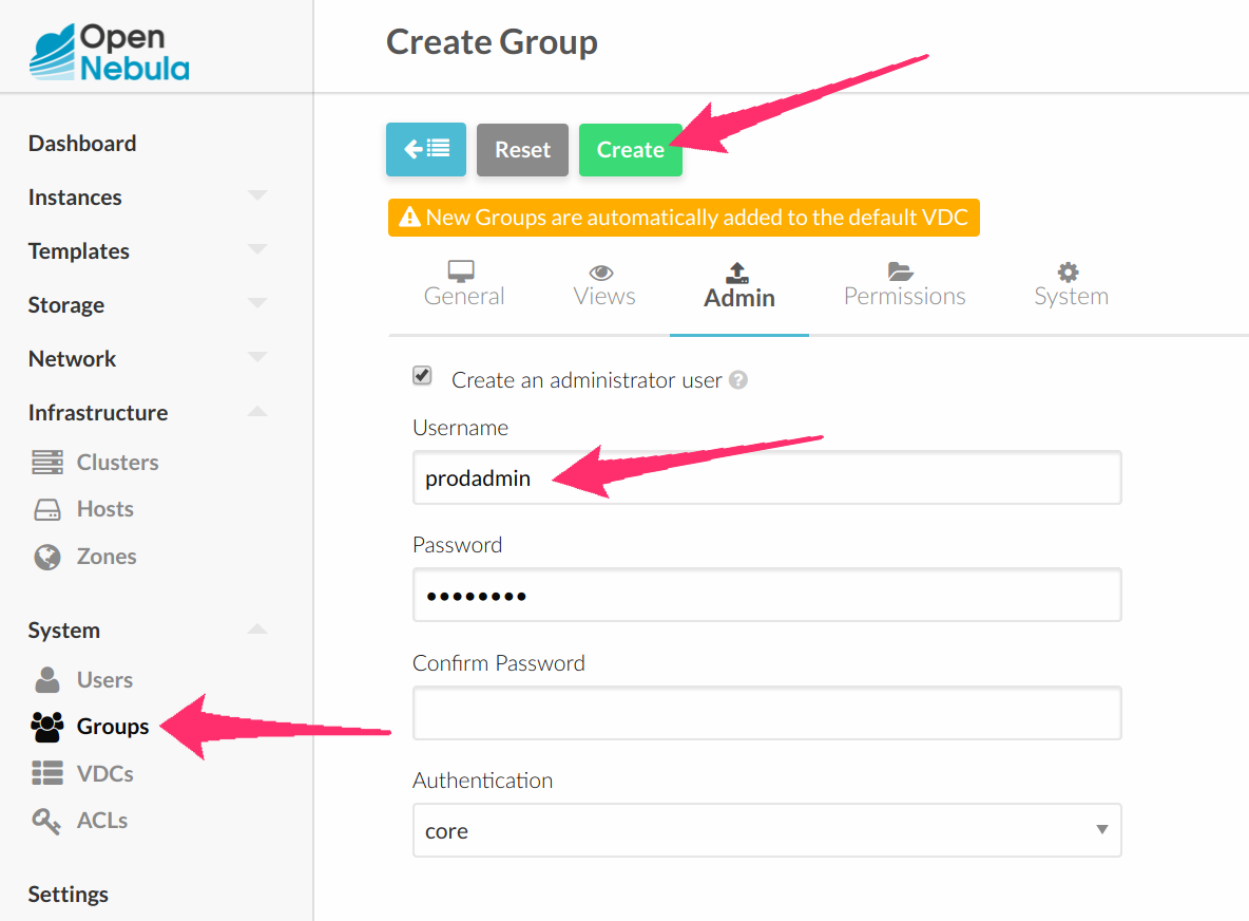
More information on available operations over VMs here.

3.4 Create a Virtual Datacenter

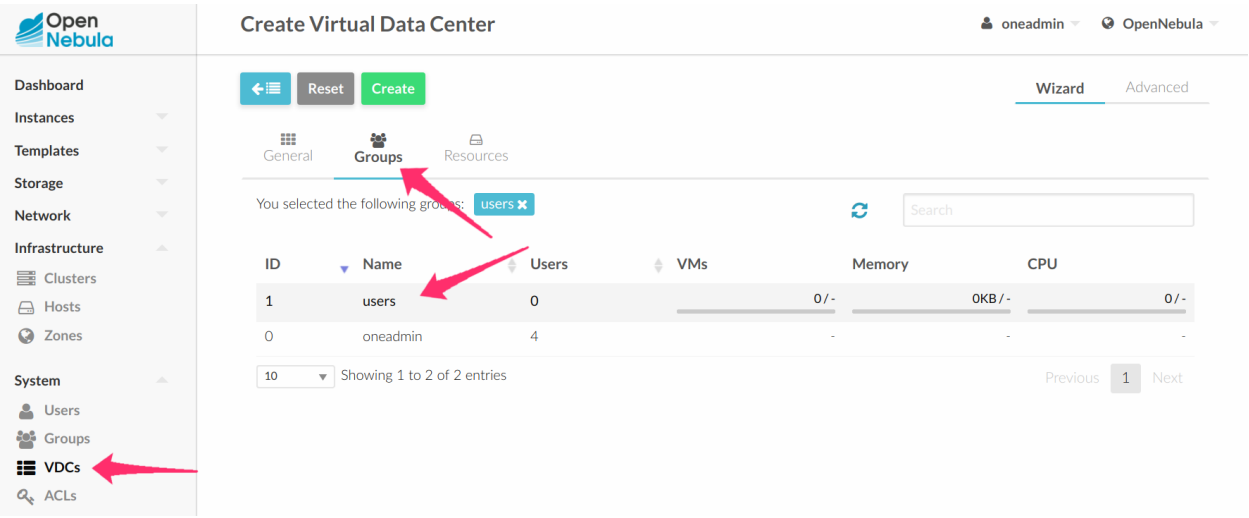
A Virtual Datacenter (VDC) defines an assignment of a pool of physical resources to one or several groups of users. This pool consists of logical groups of hosts, virtual networks and datastores from one or several clusters, which can be shared between VDCs. VDCs are a great way to partition your cloud into smaller clouds, and assign 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.

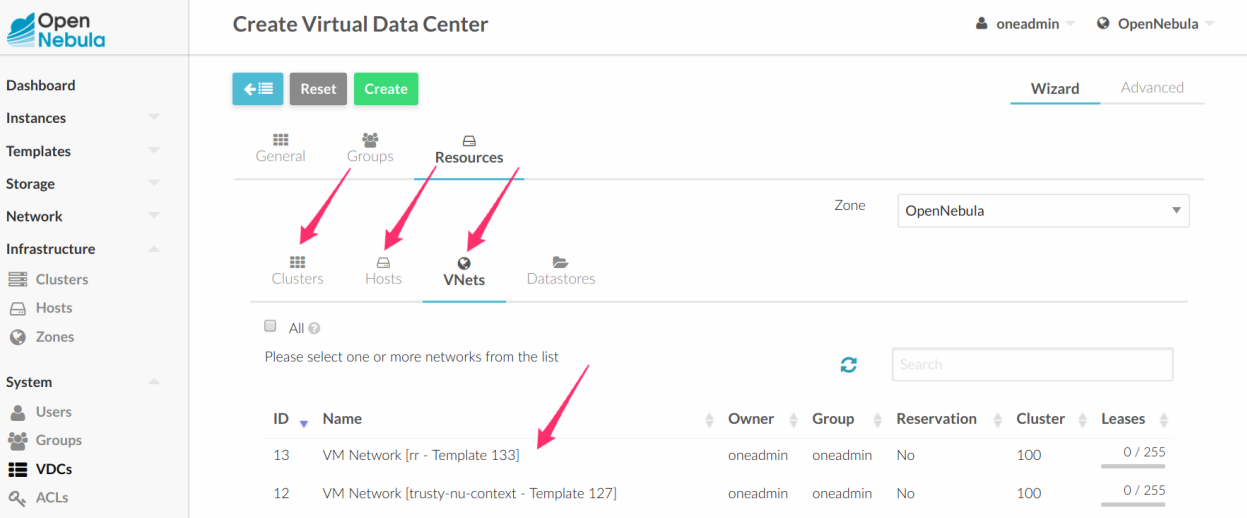
Let's create a Group (under System) named *Production* with an administrator called **prodadmin**:



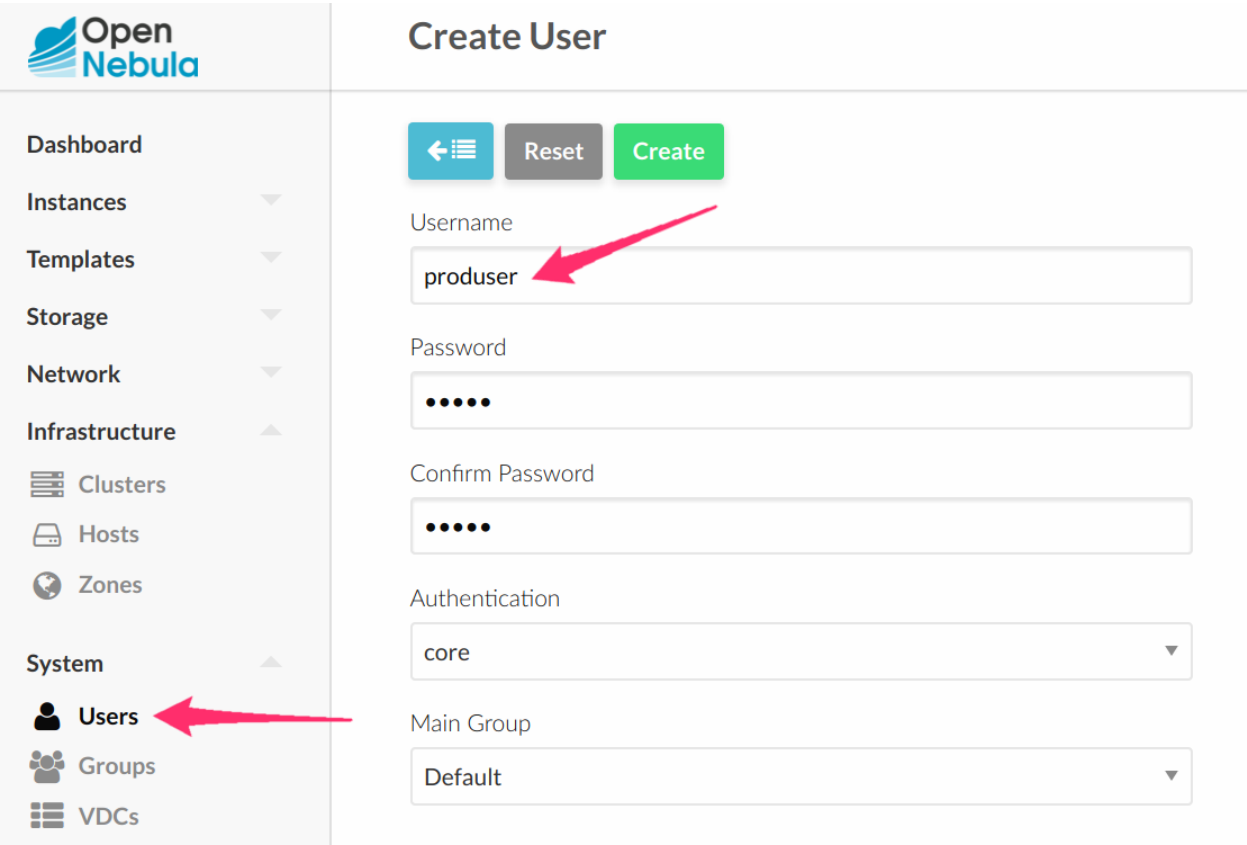
Let's create a VDCs (under System) named *ProductionVDC*, and assign the *Production* group to use it:



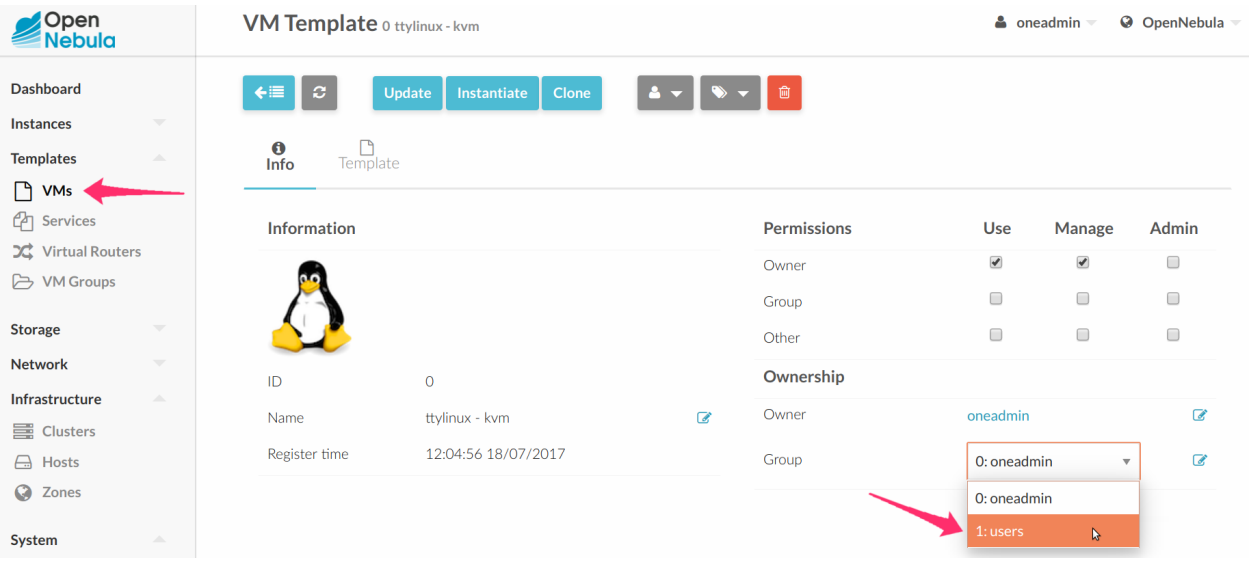
Let's add resources to the VDC under the "Resources" tab, for instance a vCenter instance and a Virtual Network:



Now login again using the newly created **prodadmin**. The Group Admin view will kick in (views will be explained later in its own section.). Try it out creating the first *produser* and assign them quotas on resource usage:

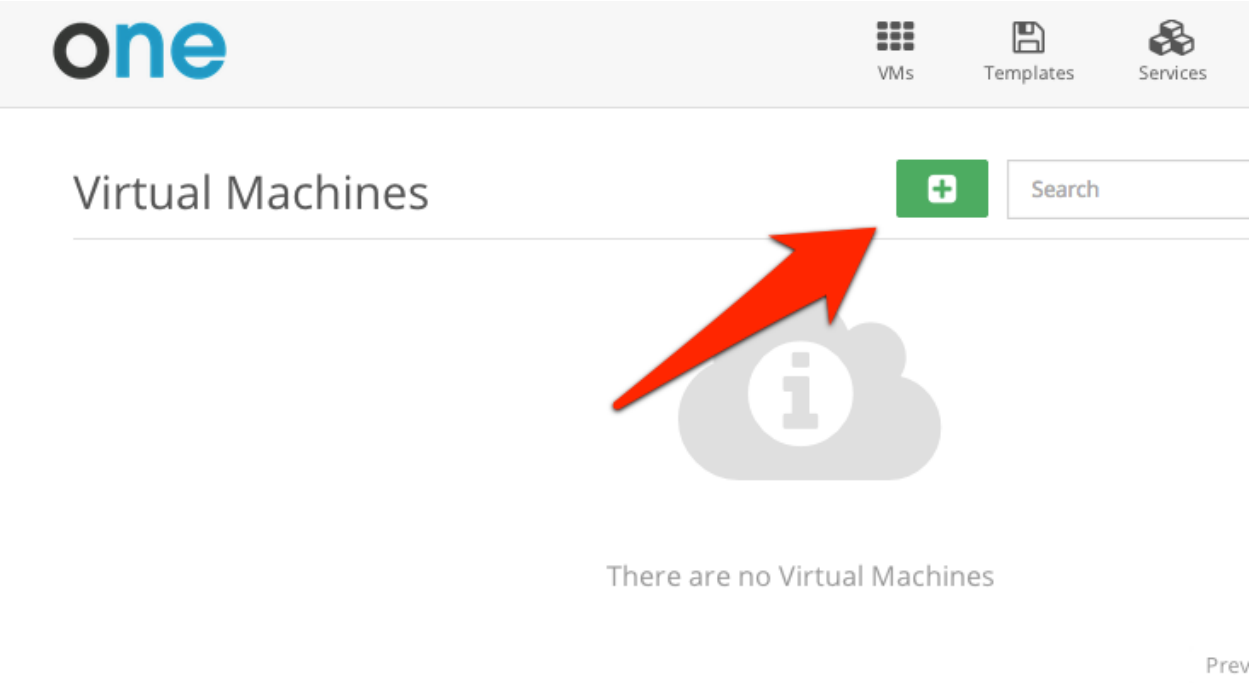


As the *CloudAdmin* user, in the vCenter View, you will be able to see all the VM Templates that have been automatically created when importing the vCenter infrastructure. You can assign any of these VM Templates to the VDC by assigned them to the Group associated to the VDC:



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 you can start consuming VMs based on the VM Template shared by the **Cloud Administrator** and allowed by the **prodadmin**:

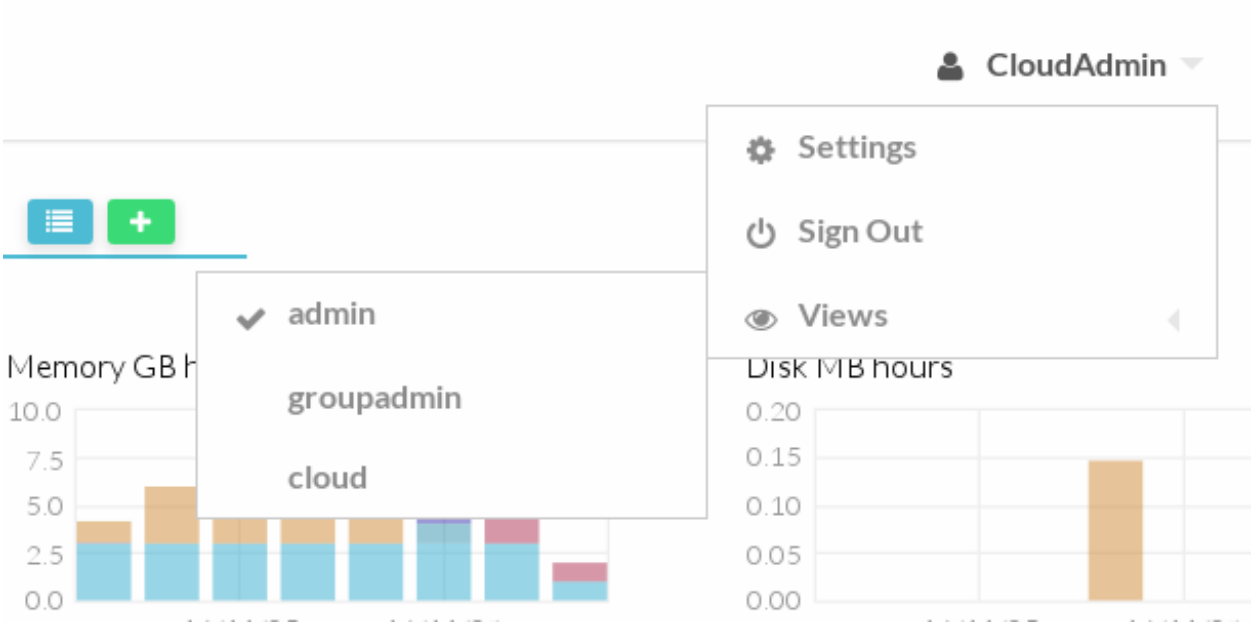


Read more about Group and VDC managing.

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.

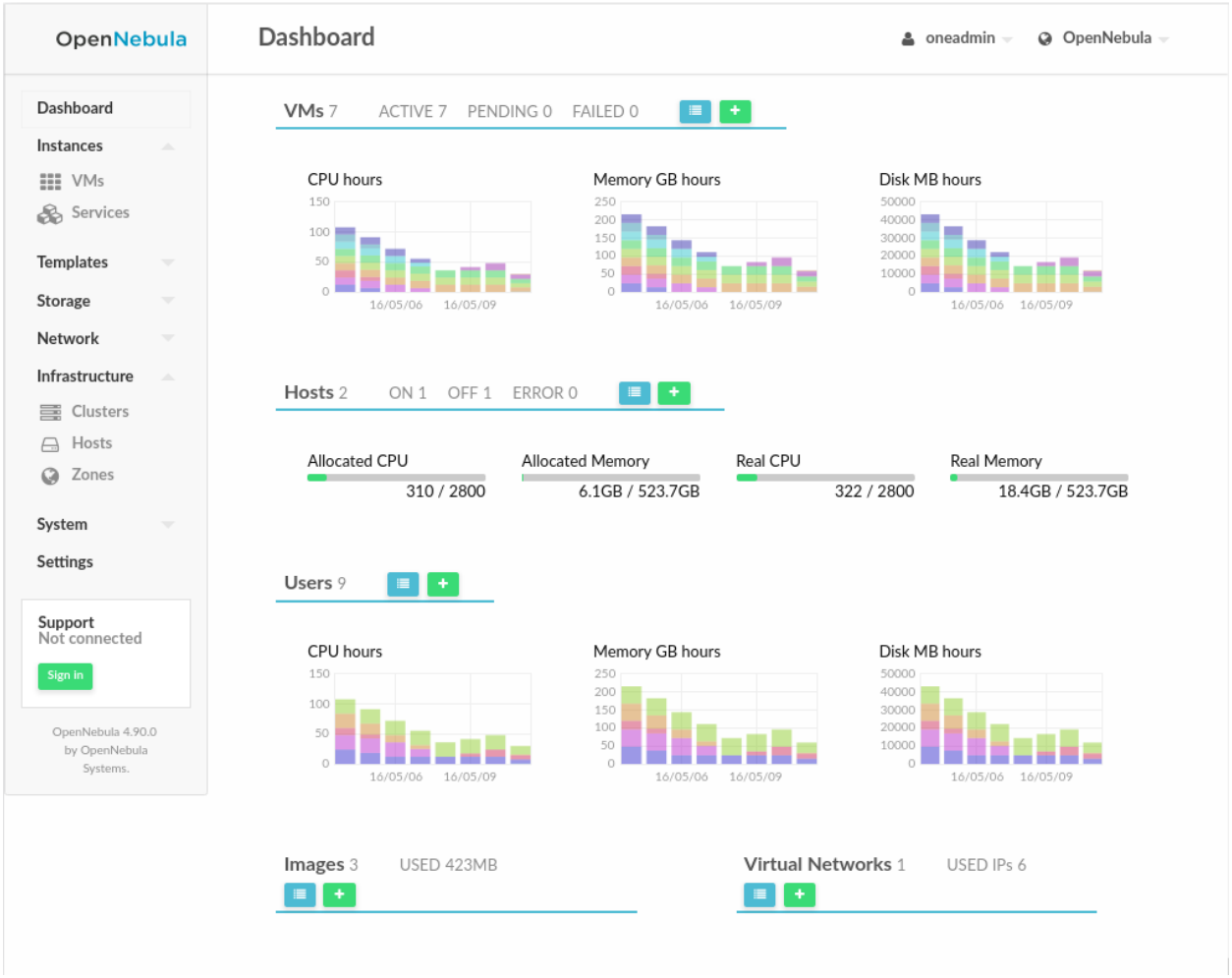
You can select one of the available views clicking in the username at the top right of the screen and selecting the Views entry.



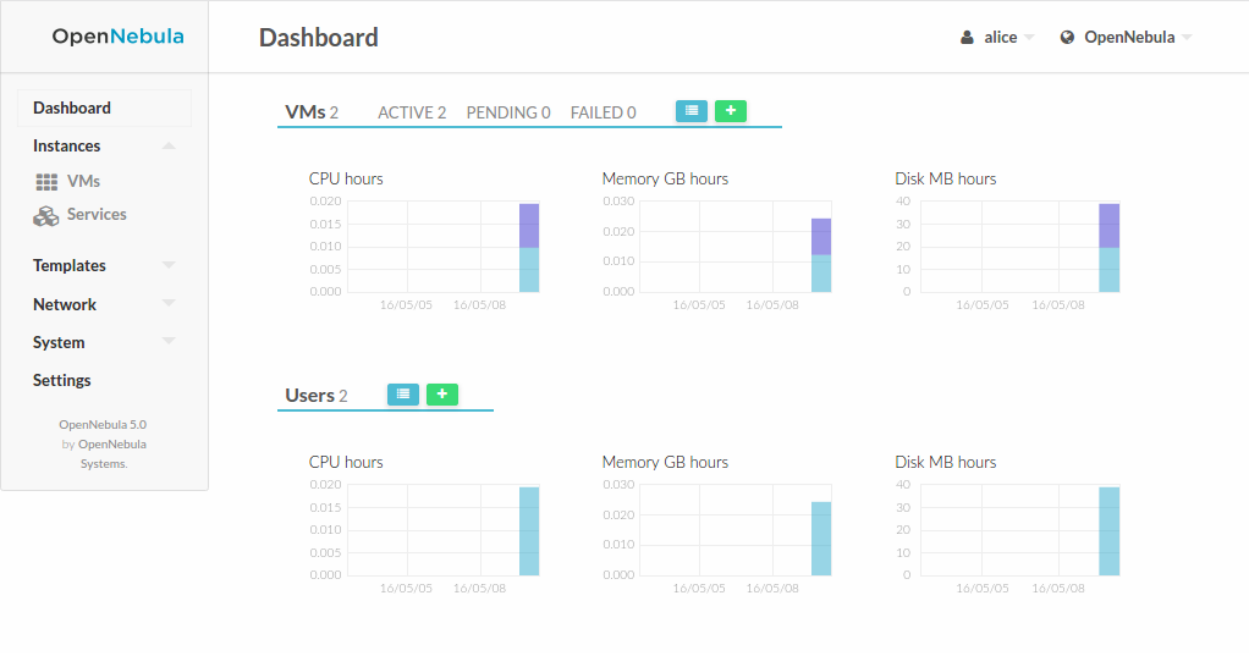
3.5.1 Web Interface (Sunstone)

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

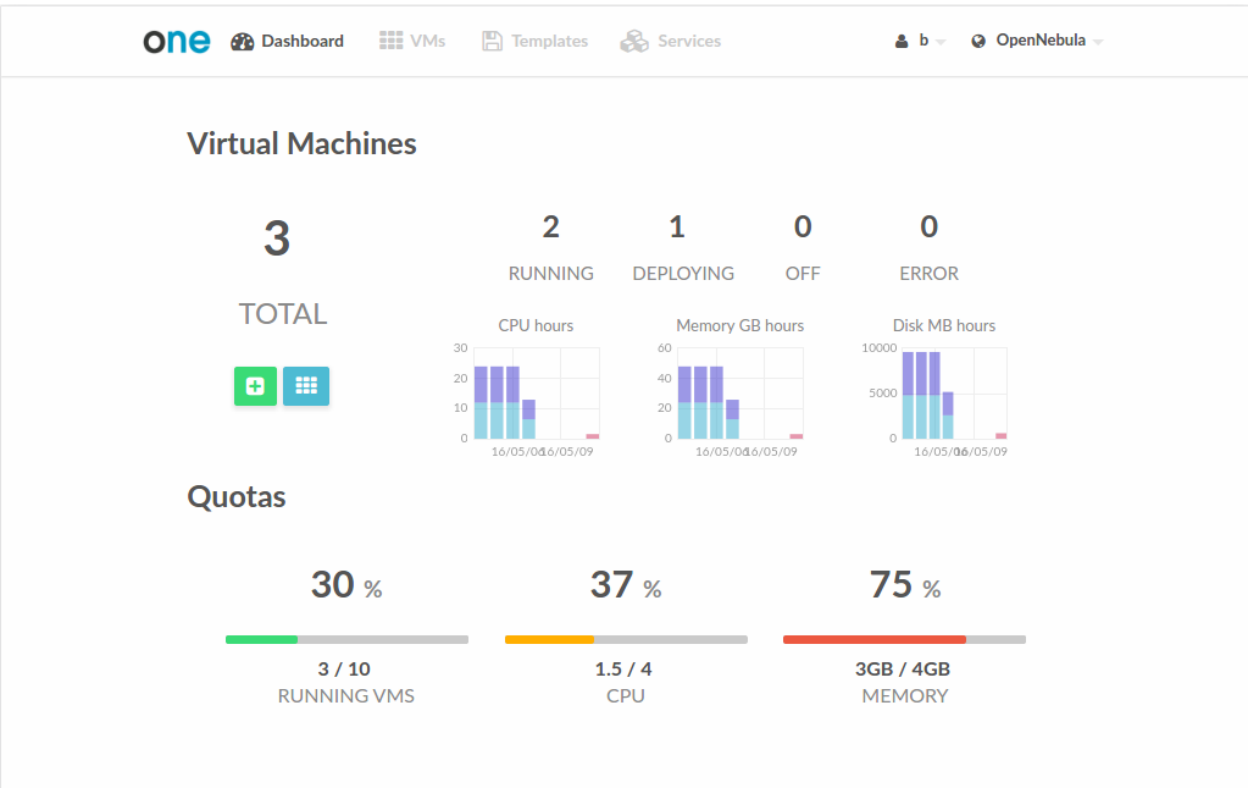
- Sunstone Admin view: Aimed at cloud administrators, this view is tailored to present all the available options to manage the physical and virtual aspects of your vCenter infrastructure.



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



- **Sunstone 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

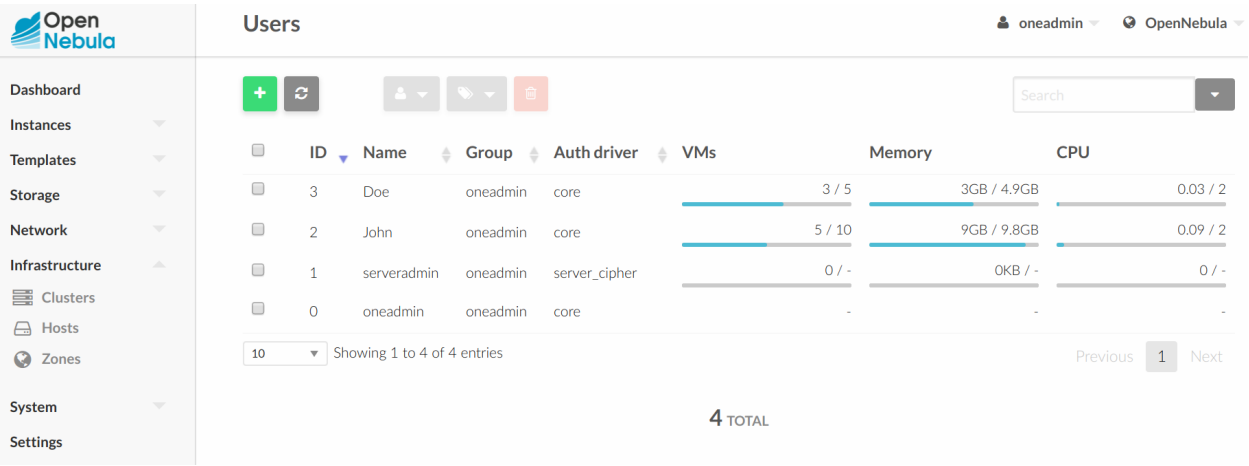
vOneCloud offers a powerful mechanism for managing, grouping and assigning roles to users. Permissions and Access Control List mechanisms ensures the ability to allow or forbid access to any resource controlled by vOneCloud, being physical or virtual.

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

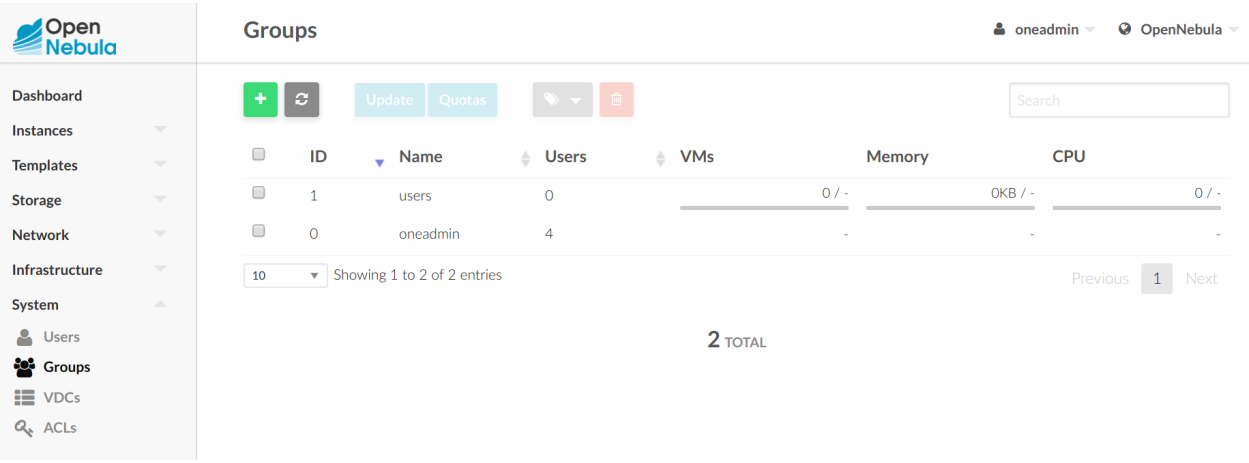


Learn more about user management here.

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).

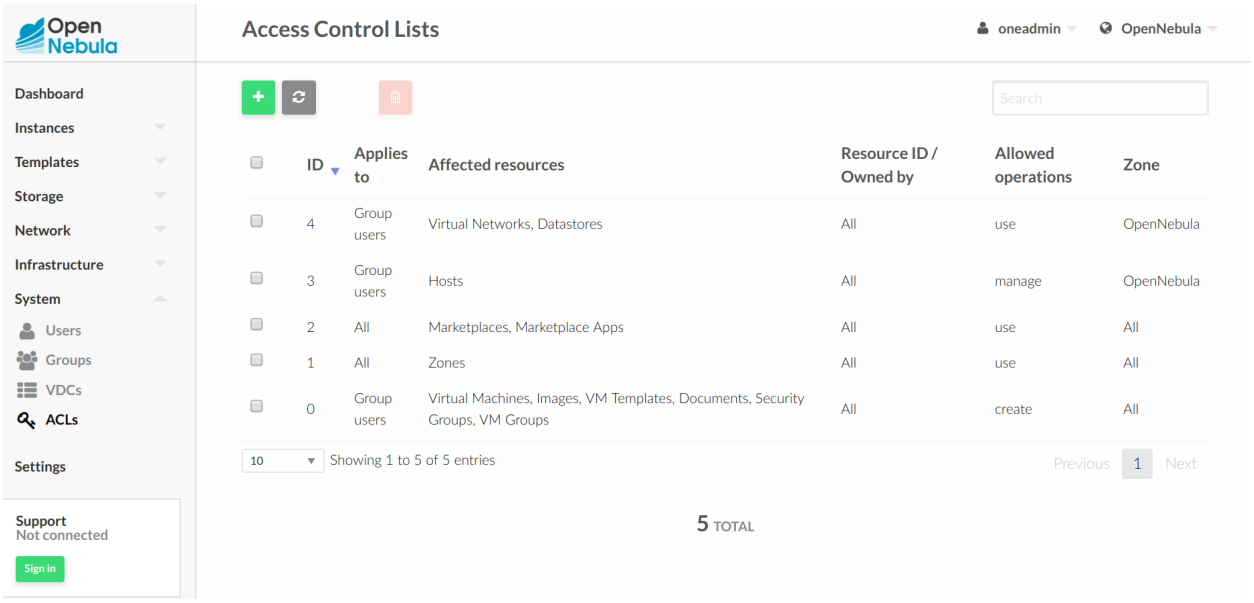
A VDC defines an assignment of one or several groups to a pool of physical resources. This pool of physical resources consists of resources from one or several clusters, which are logical groups of hosts and virtual networks. VDCs are a great way to partition your cloud into smaller clouds, and asign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

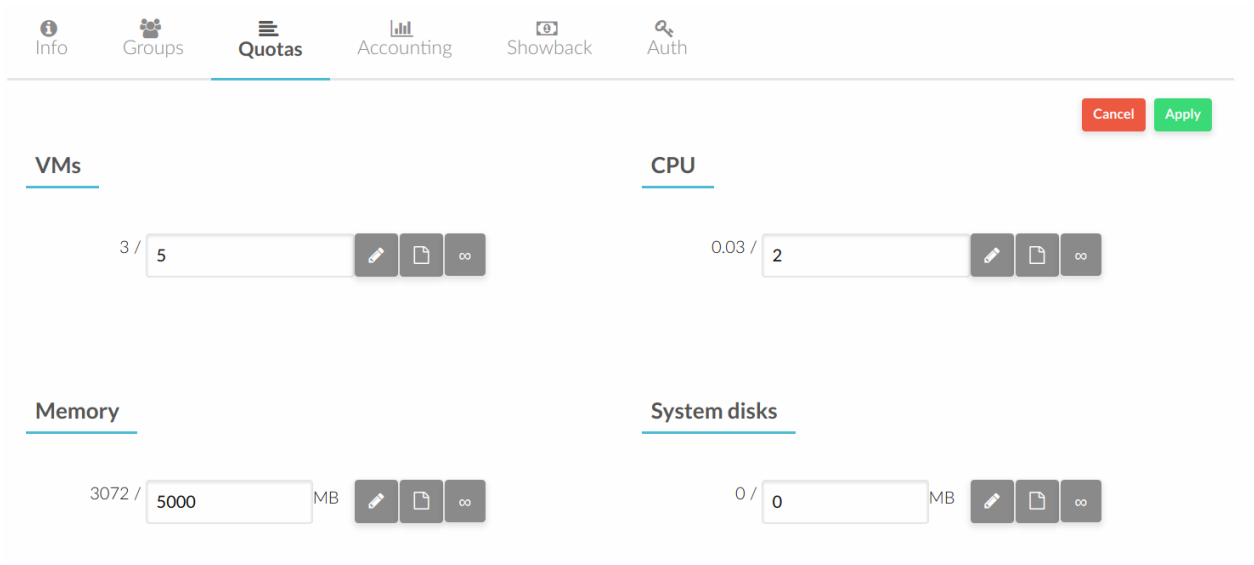
Quota limits can be set for:

- **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

Quotas can be updated either from the vCenter View or from the Group Admin View.



Refer to this guide to find out more.

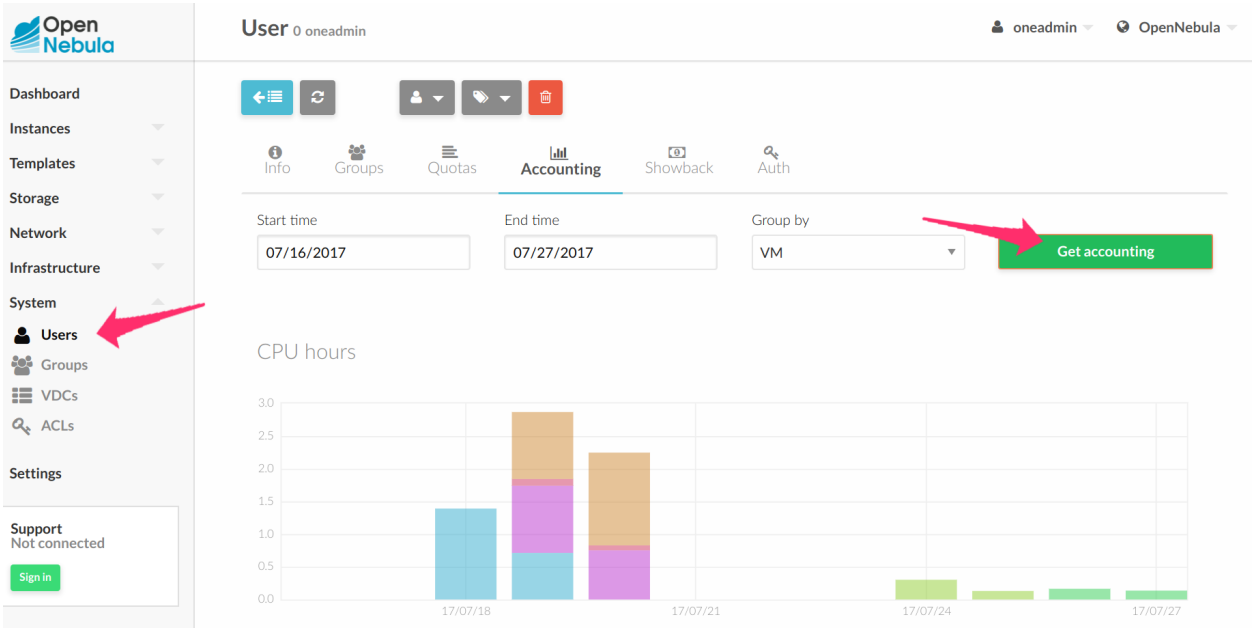
4.4 Accounting & Monitoring

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.

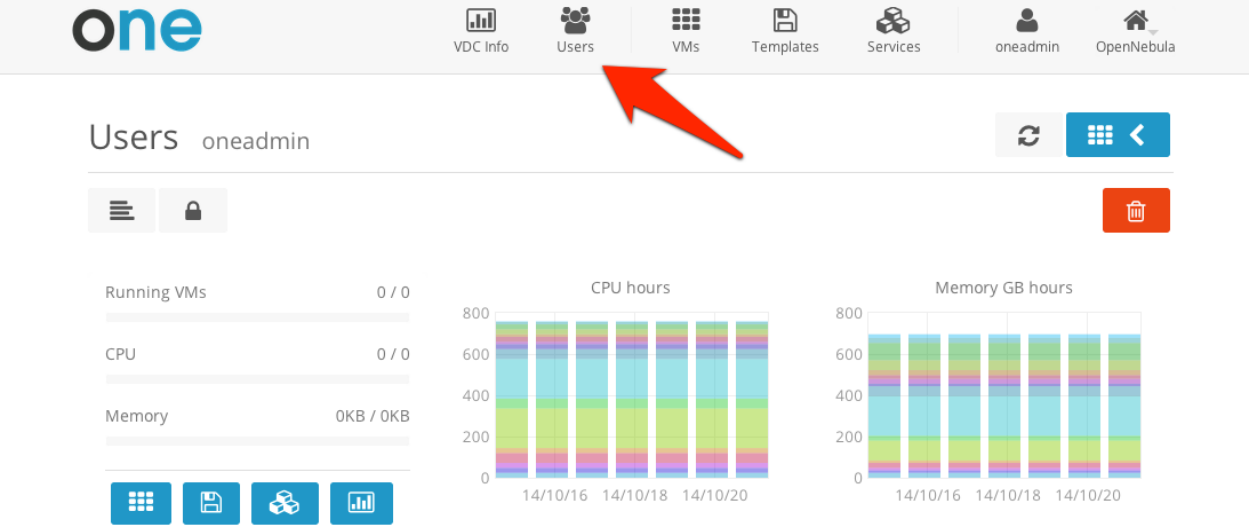
The monitoring subsystem gathers information relative to hosts and virtual machines, such as host and VM status, basic performance indicators and capacity consumption. vOneCloud comes preconfigured to retrieve such information directly from vCenter.

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.

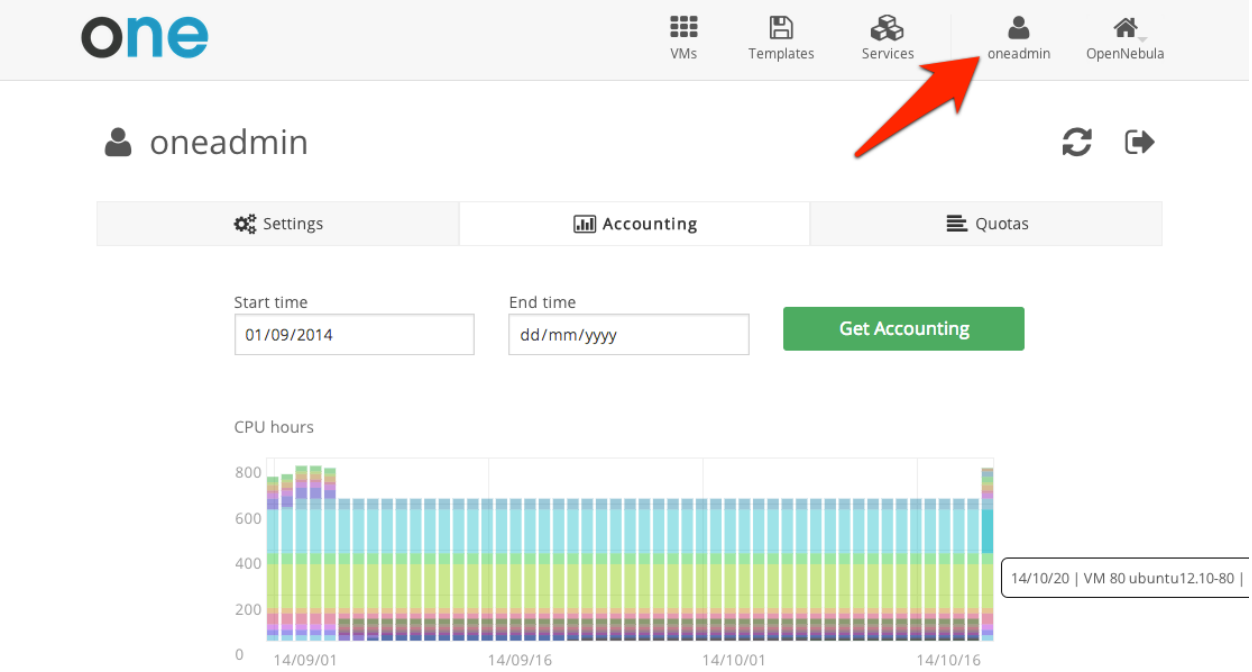
Accounting information is available from the vCenter View:



From the Group Admin View:



And from the vCenter Cloud View:



Learn more on the monitoring and accounting subsystems

4.5 Showback

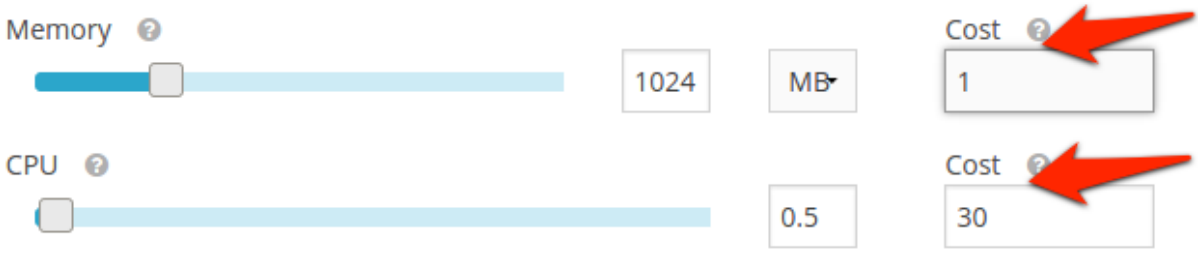
vOneCloud ships with functionality to report resource usage cost. Showback reports are genereted daily (at midnight)using the information retrieved from OpenNebula.

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

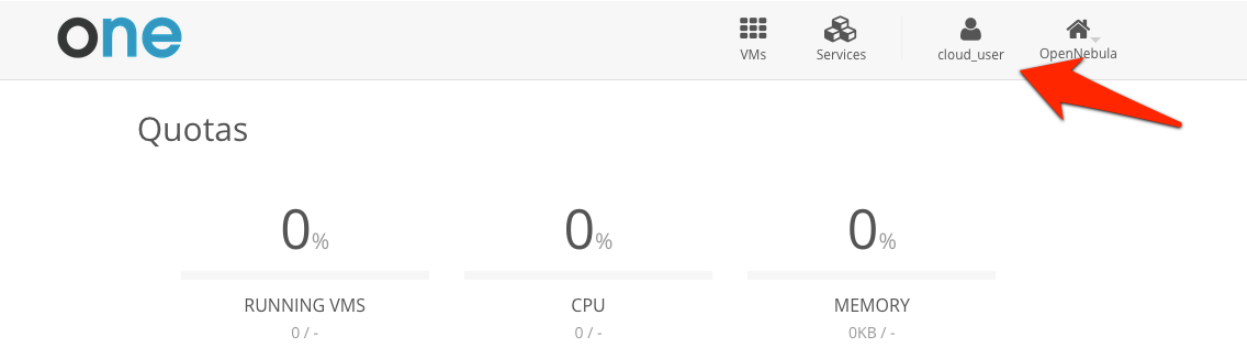
MB per hour. The cost units are abstract and their equivalent to monetary or other cost metrics have to be defined in each deployment.

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.

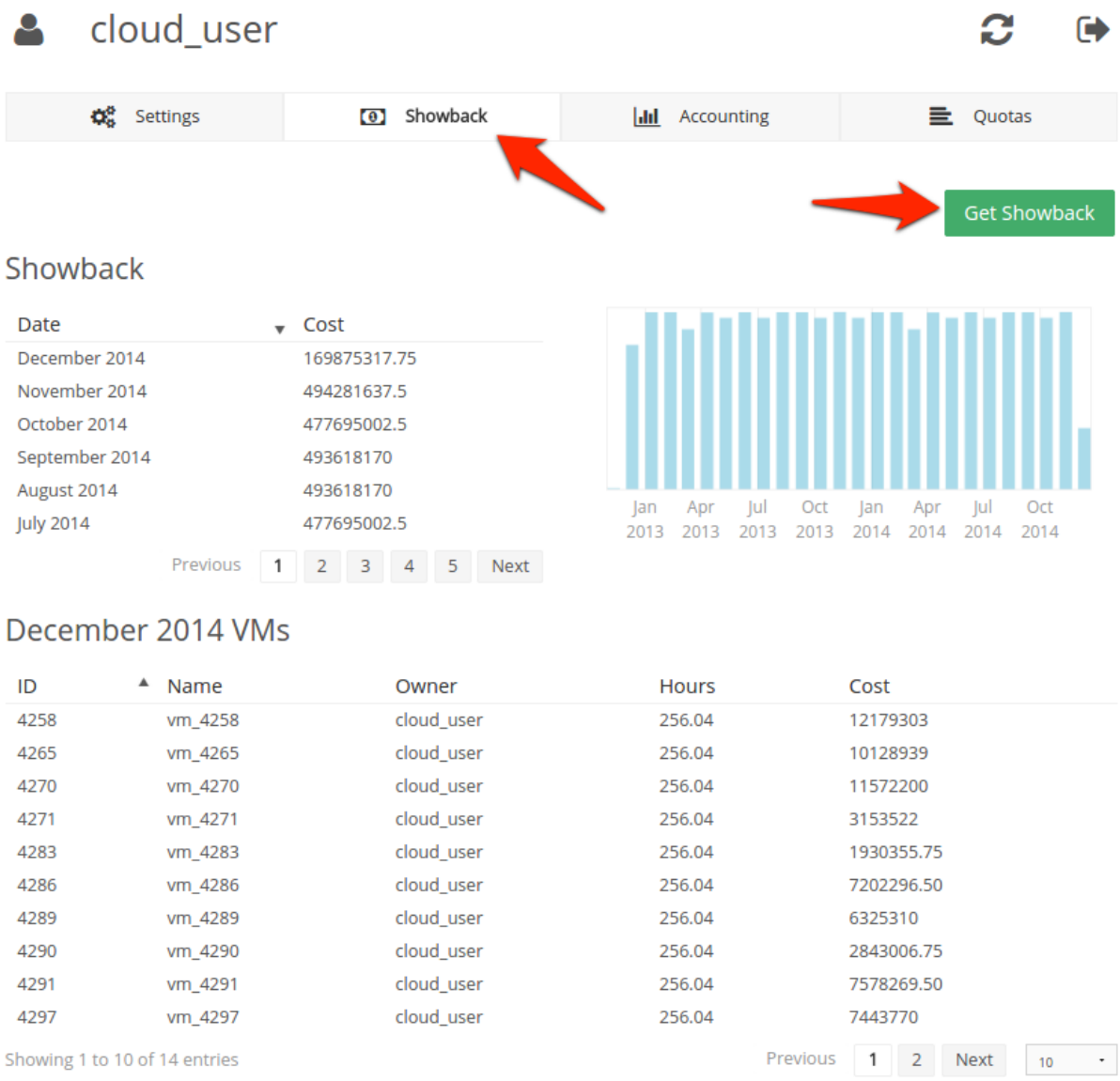


Retrieve Monthly Reports

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 provides two mechanisms to configure the newly created VMs.

- OpenNebula Contextualization: It allows configuration and information sharing between the vOneCloud interface and the Virtual Machine. This methods needs to build a a template for contextualization and to prepare the guest OS.
- *vCenter Customization Specifications*: It ties the template with a vCenter Customization Specification so it is configured on VM creation

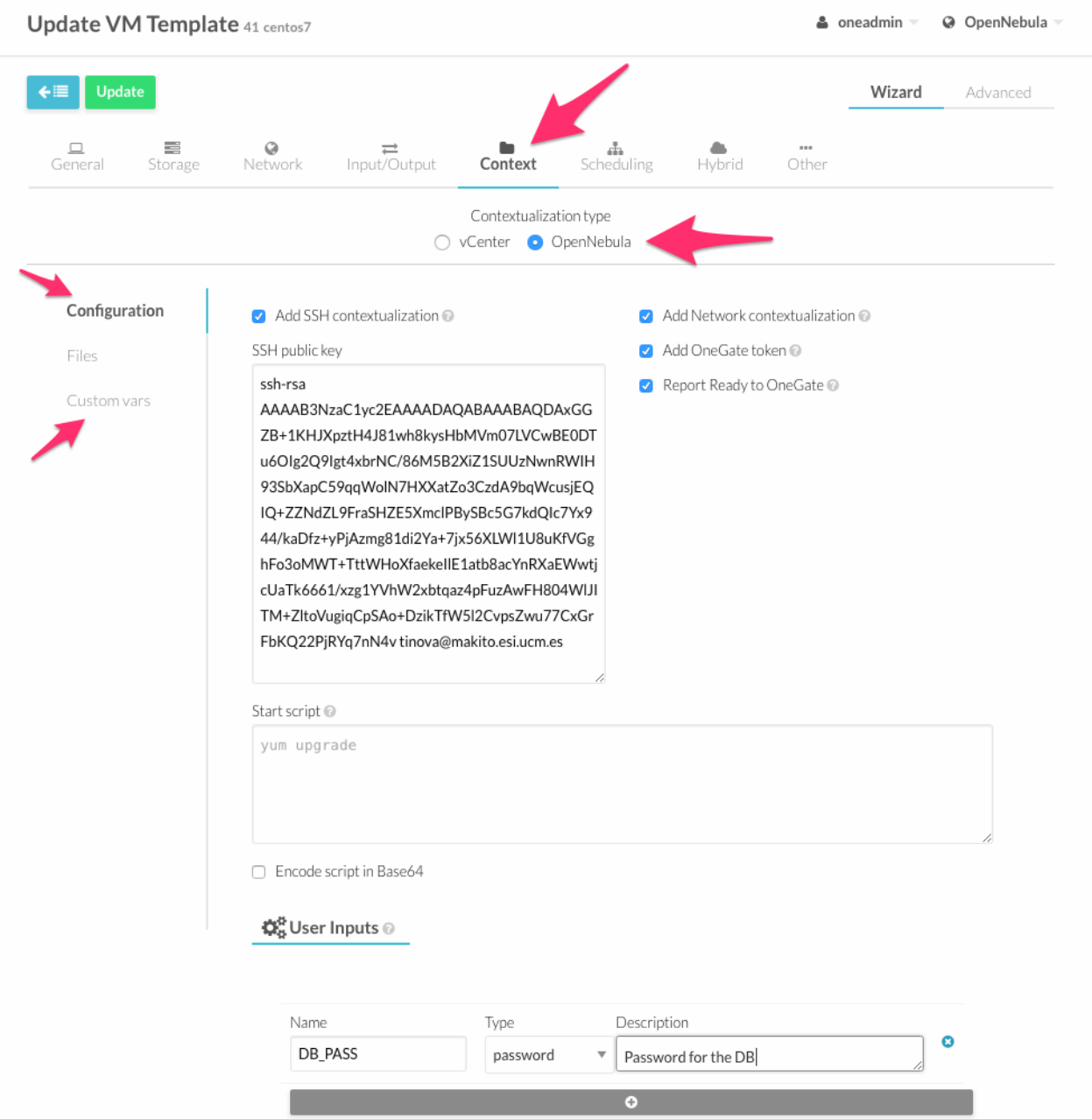
Warning: These options can not be used together. A template can use either OpenNebula Contextualization or vCenter customization.

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 vOneCloud VM 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 instantiated VM template. Make sure you follow the *Guest Contextualization* guide to properly prepare your VM templates.



Warning: Passing files to VMs through contextualization is not supported

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

5.2.1 Network Configuration

vOneCloud does not rely on a DHCP server to configure networking in the Virtual Machines. To do this configuration it injects the network information in the contextualization section. This is done checking the "Add Network configuration" check box. When vOneCloud finds this option it adds the IP information for each of the network interfaces configured plus extra information that resides in the Virtual Network template, like DNS, gateway and network mask.

The parameters used from the Virtual Network template are explained in the OpenNebula documentation, Managing Virtual Networks section.

5.2.2 User Credentials

One of the other very important things you have to configure is user credentials to connect to the newly created Virtual Machine.

For Linux base images we recommend to use SSH public key authentication and using it with vOneCloud is very convenient. The first thing the users should do its to add their SSH public key (or keys) to its vOneCloud user configuration, this can be done in the Settings section of the web interface. The Context section of the VM Template needs to have the "Add SSH contextualization" check box selected. Using this system the new Virtual Machines will be configured with the SSH public key of the user that instantiated it.

For Windows machines SSH is not available but you can use the options `USERNAME` and `PASSWORD` to create and set the password of an initial administrator, they can be set as *Custom Vars*.

5.2.3 Execute Scripts on Boot

To be able to execute commands on boot, for example, to install some software, you can use the option `Start script` text area. When this option is used a new file that contains the value of the option will be created and executed.

For Windows machines this is a PowerShell script. For Linux machines this can be any scripting language as long as it is installed in the base image and the proper shebang line is set (shell scripts don't need shebang).

In this example some commands will be executed using `bash` shell that will install the package `ntpdate` and set the time.

```
#!/bin/bash
yum update
yum install -y ntpdate
ntpdate 0.pool.ntp.org"
```

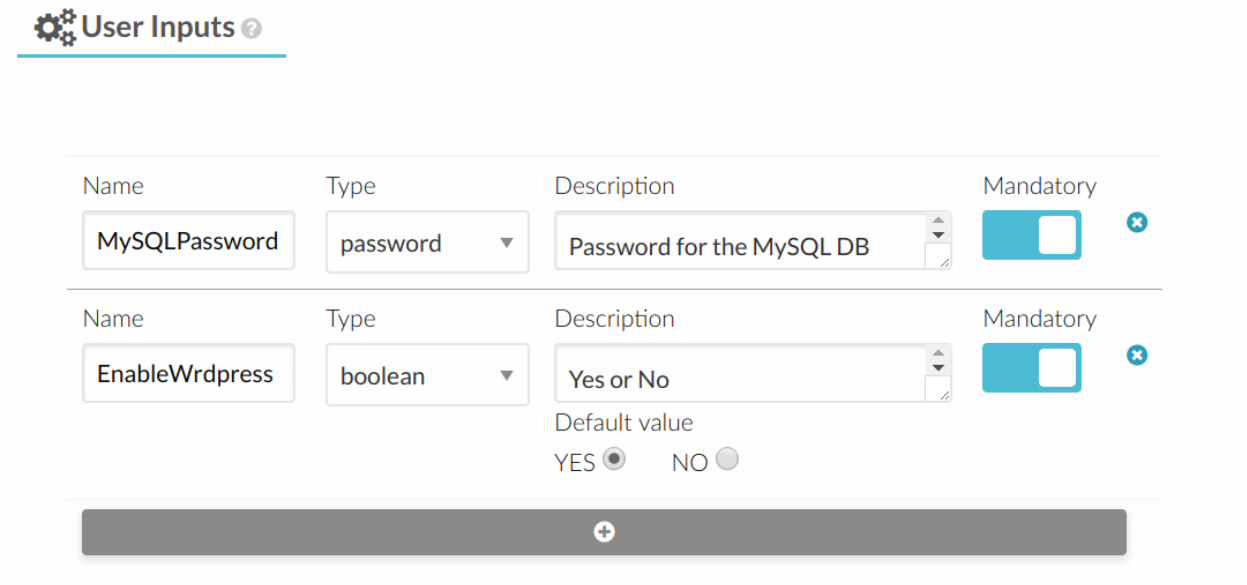
If you are using complex scripts, it is a good idea to use the "encode script in Base64" option.

There are more options that can be set in the contextualization section. You can read about them in the Virtual Machine Definition File reference section

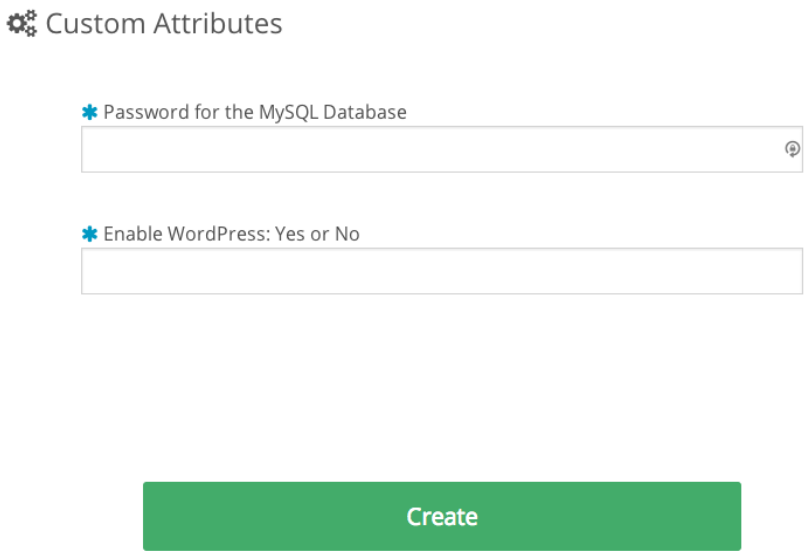
5.2.4 User Inputs

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 administrator 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



5.2.5 Custom vars

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

Note: There are more options that can be set in the contextualization section. You can read about them in the Virtual Machine Definition File reference section

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 and 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.

In Linux guests, the information can be consumed using the following command:

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

The Linux packages can be downloaded from its project page and the Windows one from its project page. The steps to prepare a contextualized VM Template are:

5.3.1 Step 1. Start a VM with the OS you want to Customize

Supported contextualization packages are available for the following OS's:

- **CentOS/RHEL** >= 6
- **Debian** >= 6
- **Ubuntu** >= 11.10
- **Windows** >= 7
- **Windows Server** >= 2008

If you already happen to have a VM or Template in vCenter with the installed OS you can start it and prepare it to be used with vOneCloud. Alternatively you can start an installation process with the OS media.

5.3.2 Step 2. Download Contextualization Packages to the VM

CentOS/RHEL

```
# CentOS 6.x
# wget https://github.com/OpenNebula/addon-context-linux/releases/download/v5.4.0/one-
↪context-5.4.0-1.el6.noarch.rpm
# CentOS 7.x
# wget https://github.com/OpenNebula/addon-context-linux/releases/download/v5.4.0/one-
↪context-5.4.0-1.el7.noarch.rpm
```

Debian/Ubuntu

```
# wget https://github.com/OpenNebula/addon-context-linux/releases/download/v5.4.0/one-
↪context_5.4.0-1.deb
```

Windows

Download and install the MSI package (preferred way) into `C:\`: https://github.com/OpenNebula/addon-context-windows/releases/download/v5.4.0/one-context-5.4.0.msi

Or download particular contextualization scripts to `C:\`:

- https://raw.githubusercontent.com/OpenNebula/addon-context-windows/v5.4.0/context.ps1
- https://raw.githubusercontent.com/OpenNebula/addon-context-windows/v5.4.0/startup.vbs

5.3.3 Step 3. Install Contextualization Packages and Dependencies

CentOS/RHEL 6

```
# rpm -Uvh one-context*rpm
# yum install -y epel-release
# yum install ruby # only needed for onegate command
# yum install -i dracut-modules-growroot
# dracut -f
```

CentOS/RHEL 7

```
# rpm -Uvh one-context*rpm
# yum install -y epel-release
# yum install ruby # only needed for onegate command
# yum install -y cloud-utils-growpart
```

Debian/Ubuntu

```
# dpkg -i one-context*deb
# apt-get install ruby # only needed for onegate command
# apt-get install -y cloud-utils
```

Windows

- Open the Local Group Policy Dialog by running `gpedit.msc`.
- Go to *Computer Configuration* -> *Windows Settings* -> *Scripts* -> *startup* (right click).
- Browse to the `startup.vbs` file and enable it as a startup script.

5.3.4 Step 4. Install VMware Tools

CentOS

```
# yum install open-vm-tools
```

Debian/Ubuntu

```
# apt-get install open-vm-tools
```

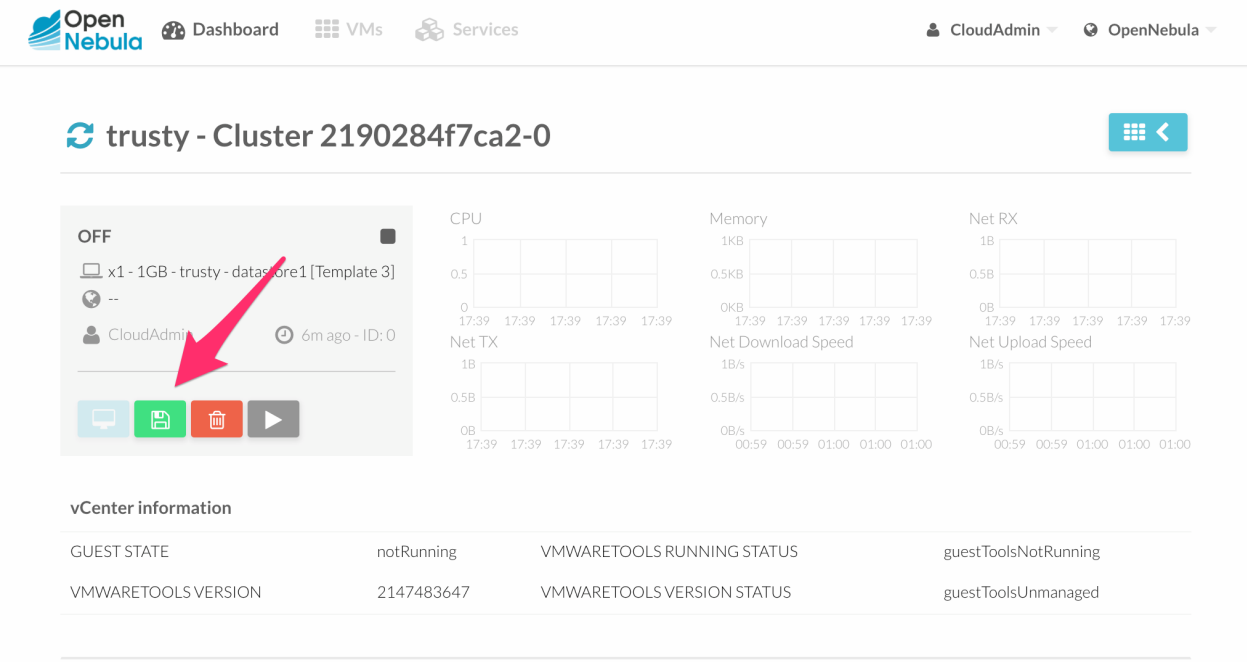
Windows

In vCenter open the VM menu, go to "Guest OS" section, click in "Install VMware Tools..." and follow the instructions.

5.3.5 Step 5. Power Off the Machine and Save it

These are the steps needed to finish the process:

- Power off the machine so it is in a consistent state the next time it boots
- Click on the Save As Template button in the Cloud View



Alternatively use the *instantiate as persistent* functionality for this step, that will create the new VM Template as soon as you terminate the VM.

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. vOneCloud vCenter drivers offers a way to tie one vOneCloud 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.

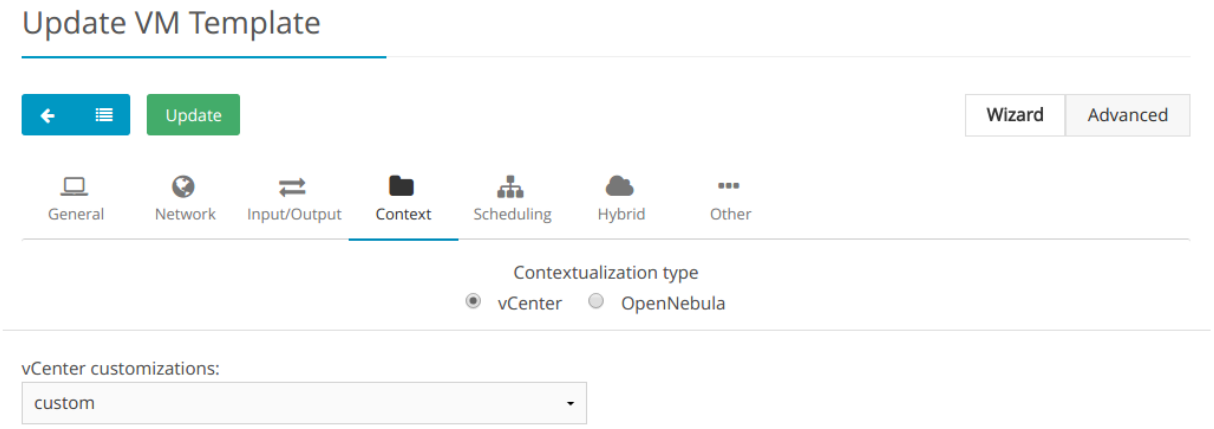
- VM network configuration must be done externally to OpenNebula. Either with a DHCP server or manually setting IPs for each interface.
- This method can be used in all the Guest OSs supported by vCenter.

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 drop down 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 vCenter

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



CHAPTER

SIX

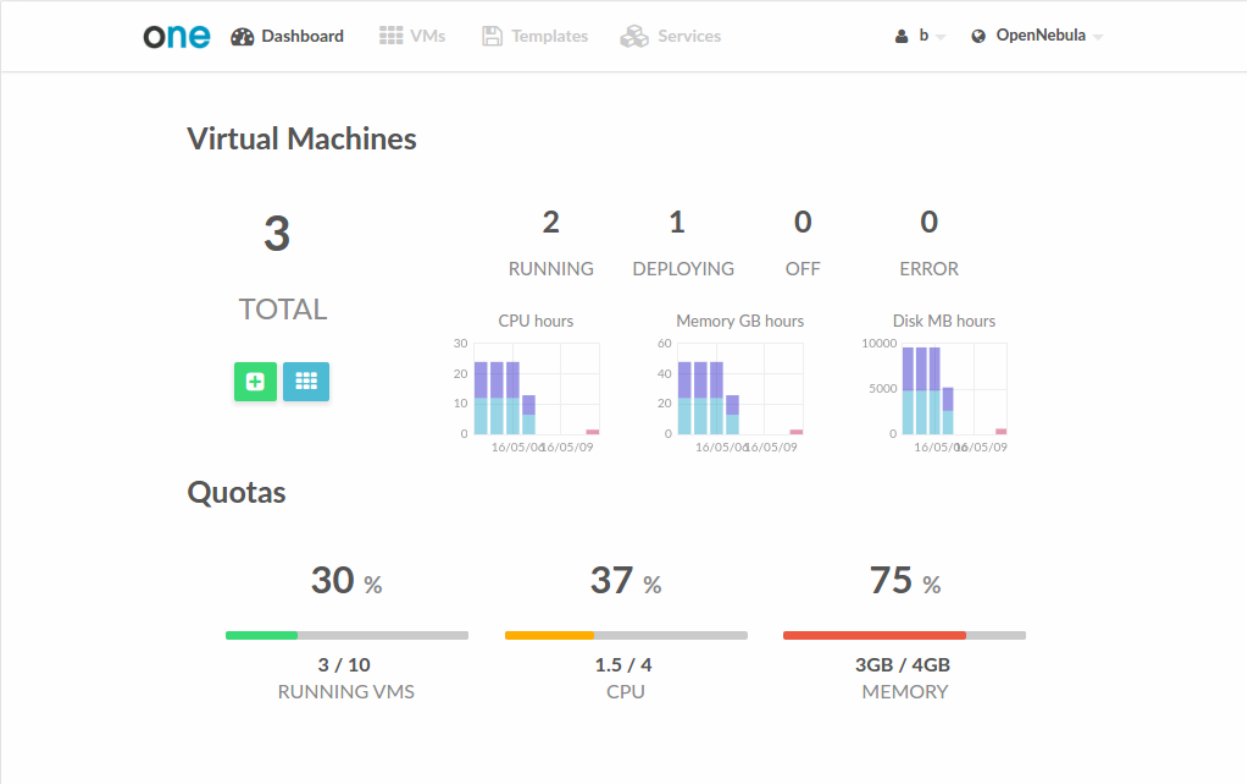
CLOUD END USER

6.1 Introduction

vOneCloud offers a simple yet powerful self-service interface for end-users of the cloud, that allows them to control the life cycle of virtual resources with a slick, functional and responsive interface. Also, Group Admins have a tailored view to manage their users and assign them appropriate virtual resources.

6.2 Self-service Cloud View

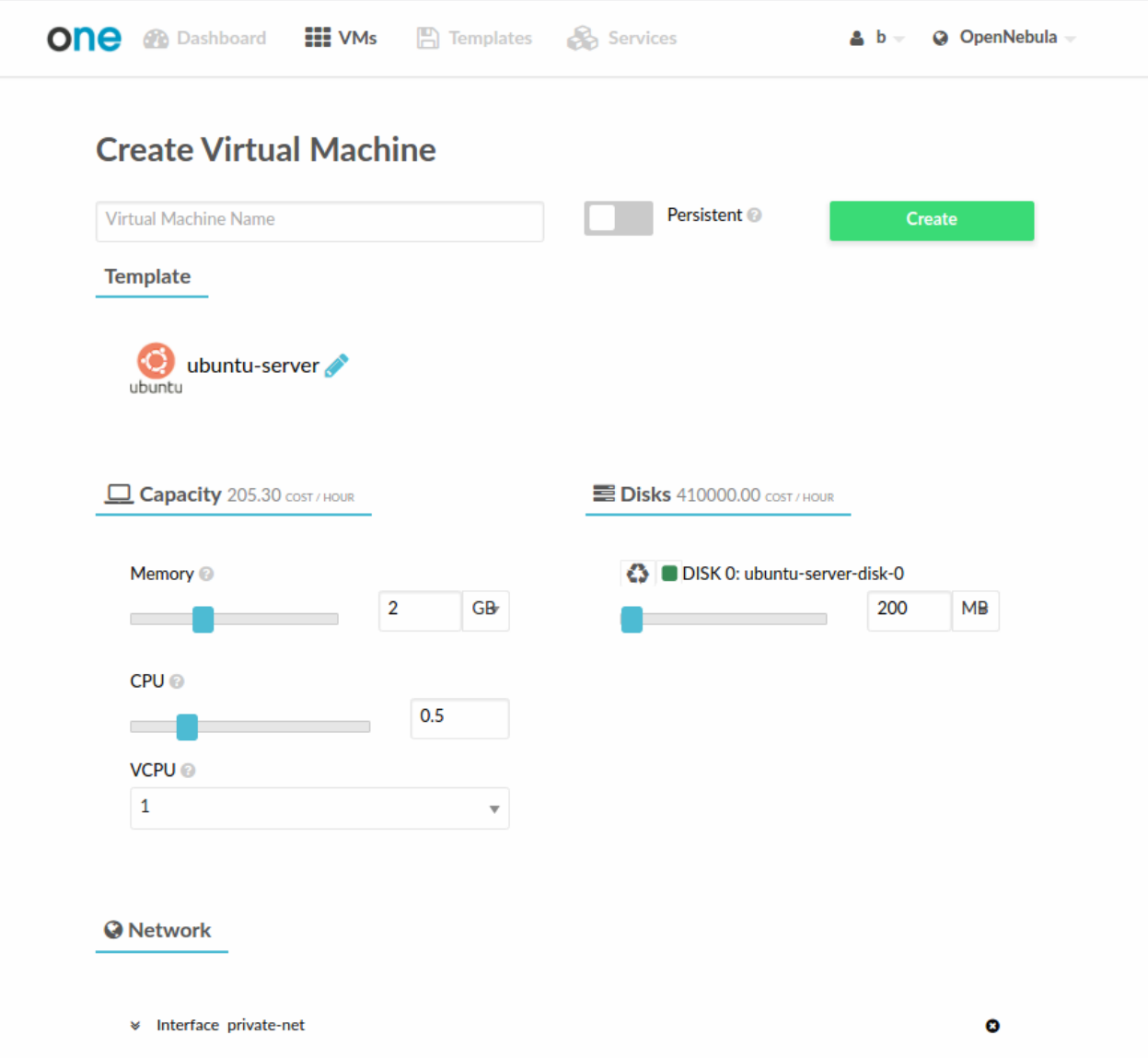
This is a simplified view intended for cloud consumers that just require a portal where they can provision new virtual machines easily. To create new VMs and Services, they just have to select one of the available templates prepared by the administrators.



6.2.1 Using the Cloud

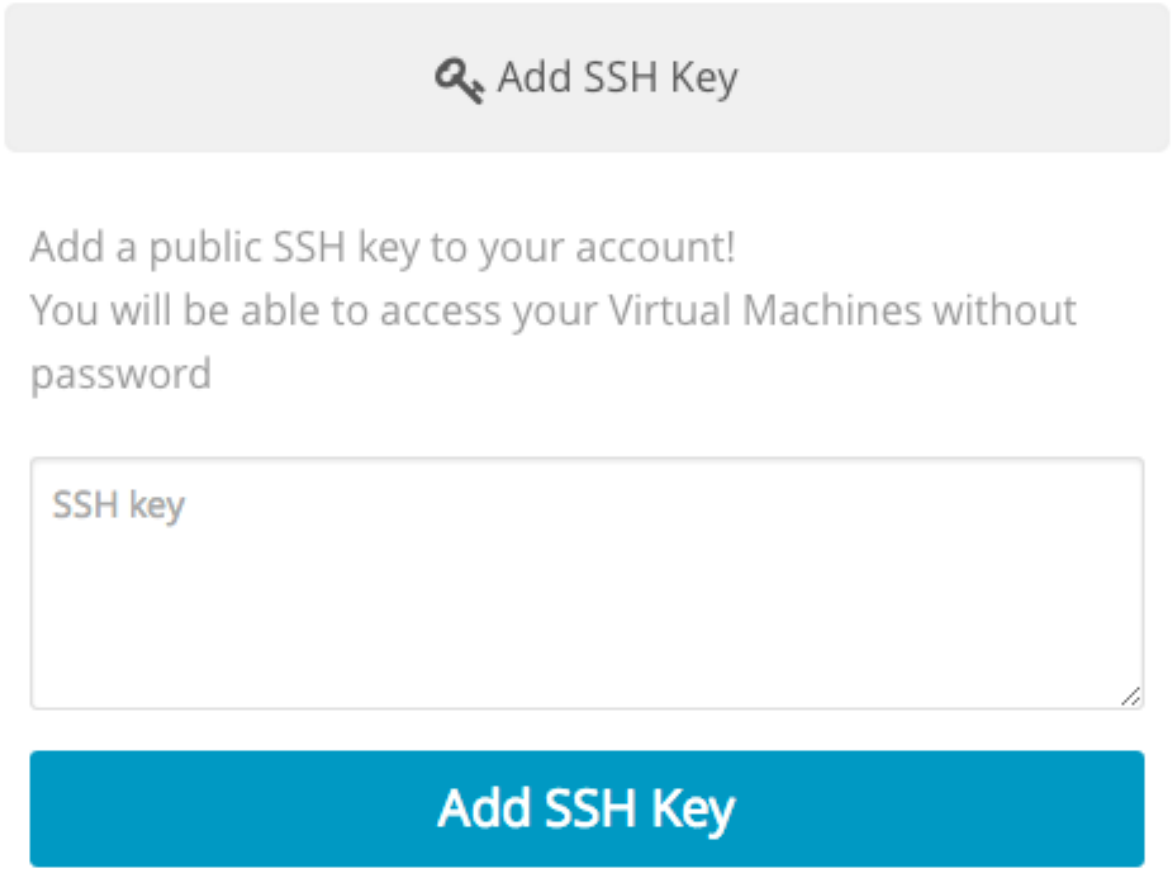
Create VM

In this scenario the cloud administrator must prepare a set of templates and images and make them available to the cloud users. These VM Templates must be ready to be instantiated, i.e. they define all the mandatory attributes. Before using them, users can optionally customize the VM capacity, resize disks, add new network interfaces and provide values required by the template.



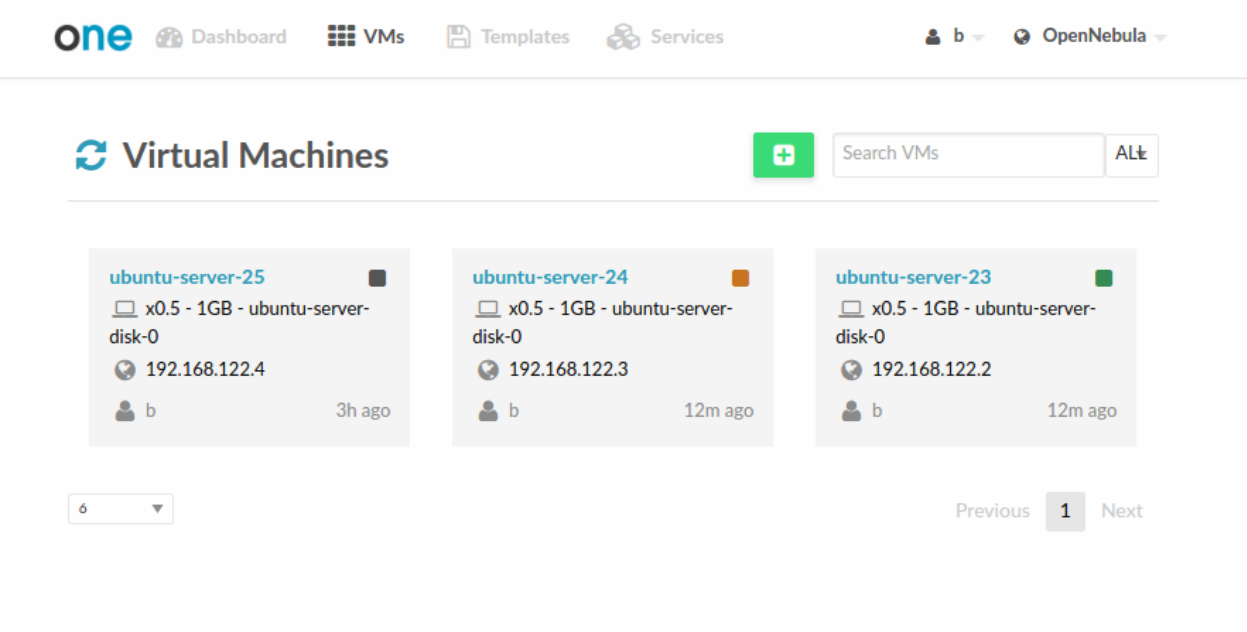
Access the VMs with SSH Keys

Any user can provide his own ssh public key to be included in the VMs created through this view. Note that the template has to be configured to include it. User can provide the ssh key clicking on its name in the top right of the screen, selecting “Settings” and then “Add SSH Key”.

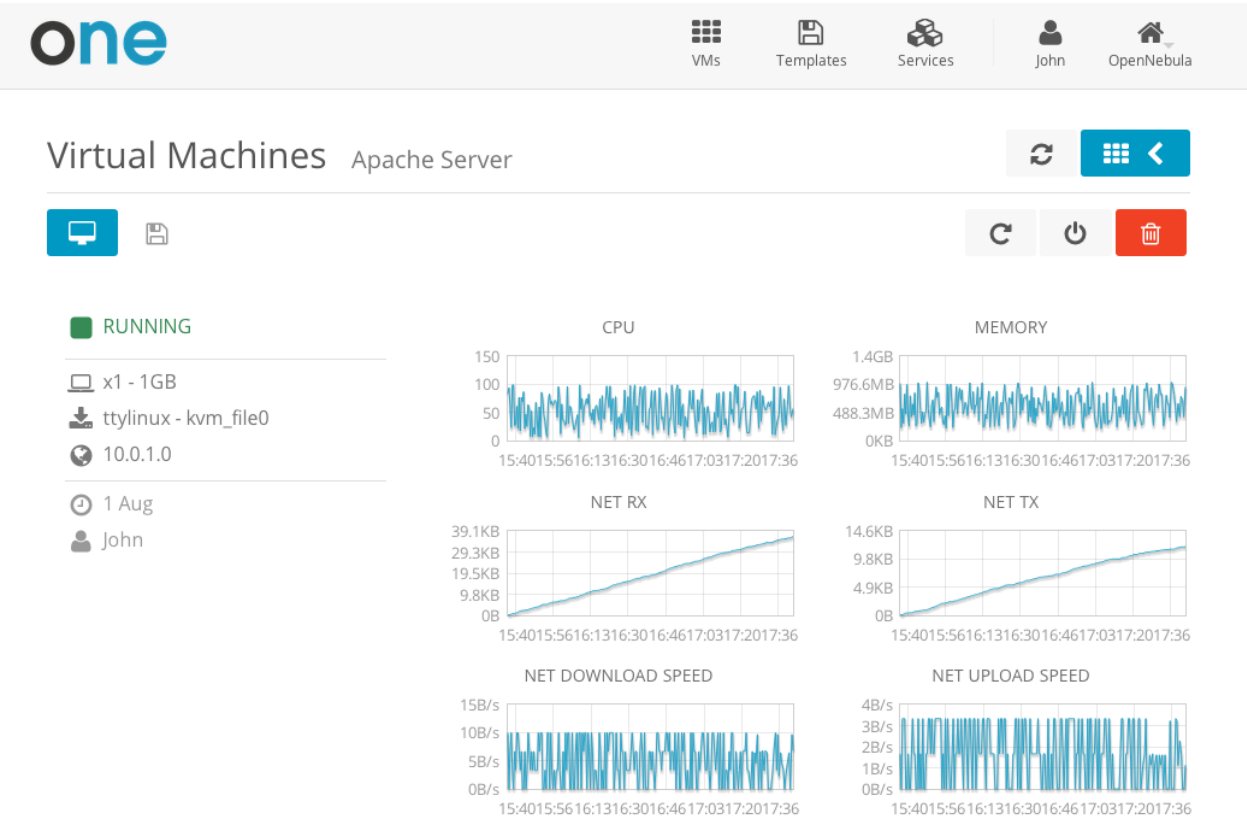


Manage VMs

The status of the VMs can be monitored from the VMs tab.



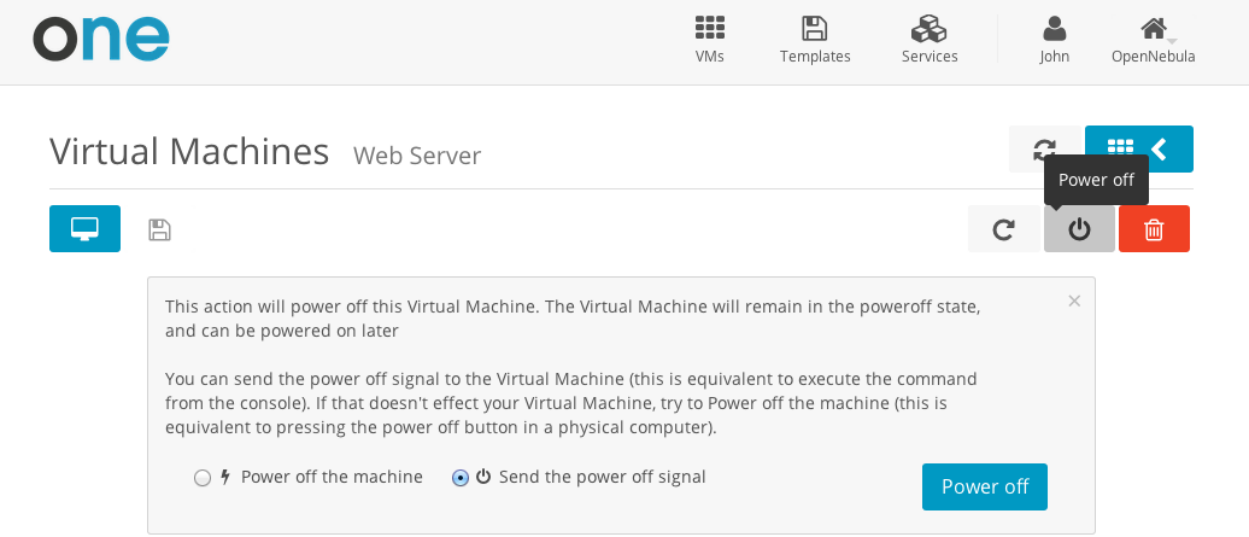
Information about the capacity, OS, IPs, creation time and monitoring graphs for a specific VM are available in the detailed view of the VM



A user can perform the following actions from this view:

- Access the VNC console, note that the Template has to be configured for this

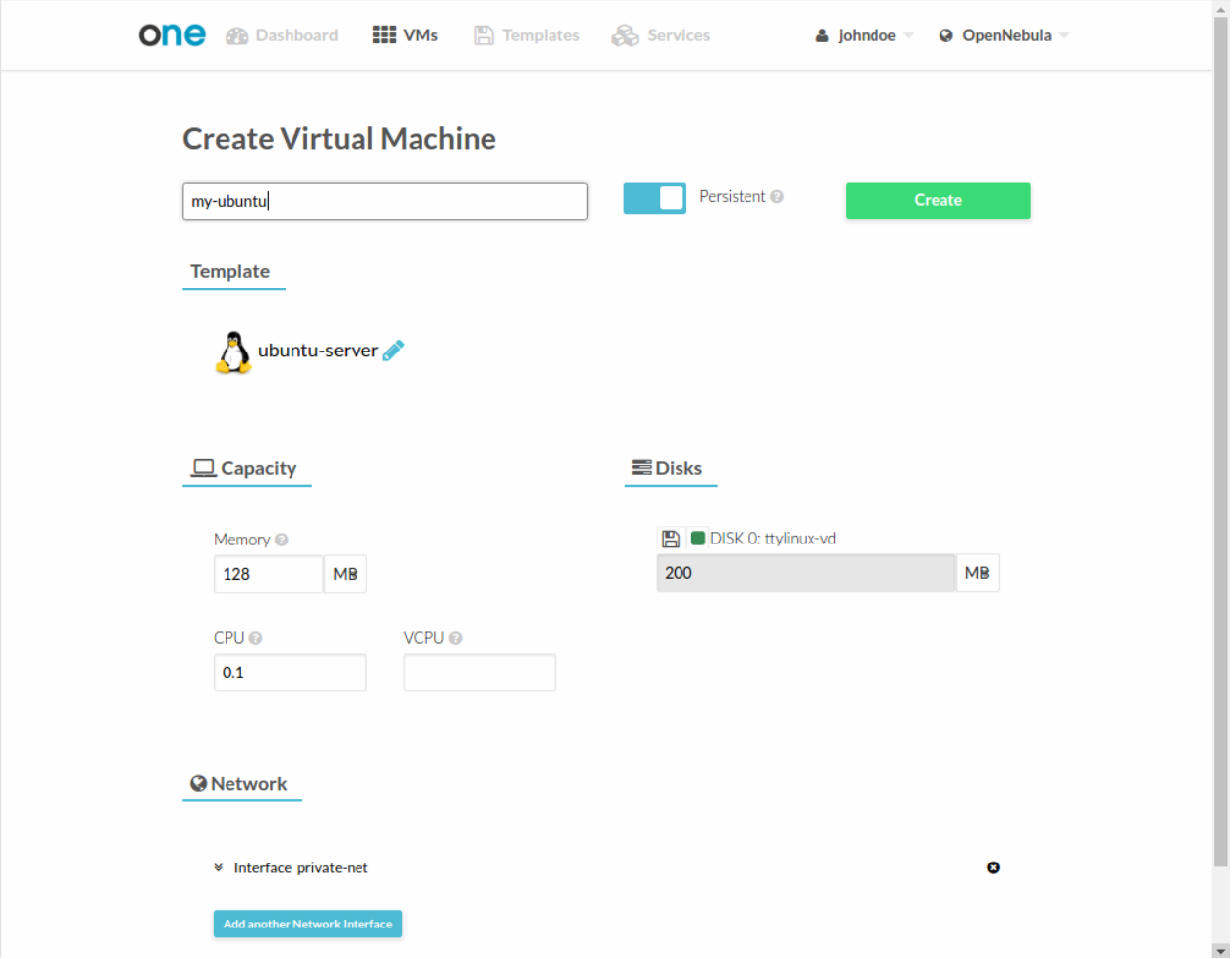
- Reboot the VM, the user can send the reboot signal (reboot) or reboot the machine (reboot hard)
- Power off the VM, the user can send the power off signal (poweroff) or power off the machine (poweroff hard)
- Terminate the VM
- Save the VM into a new Template
- Power on the VM



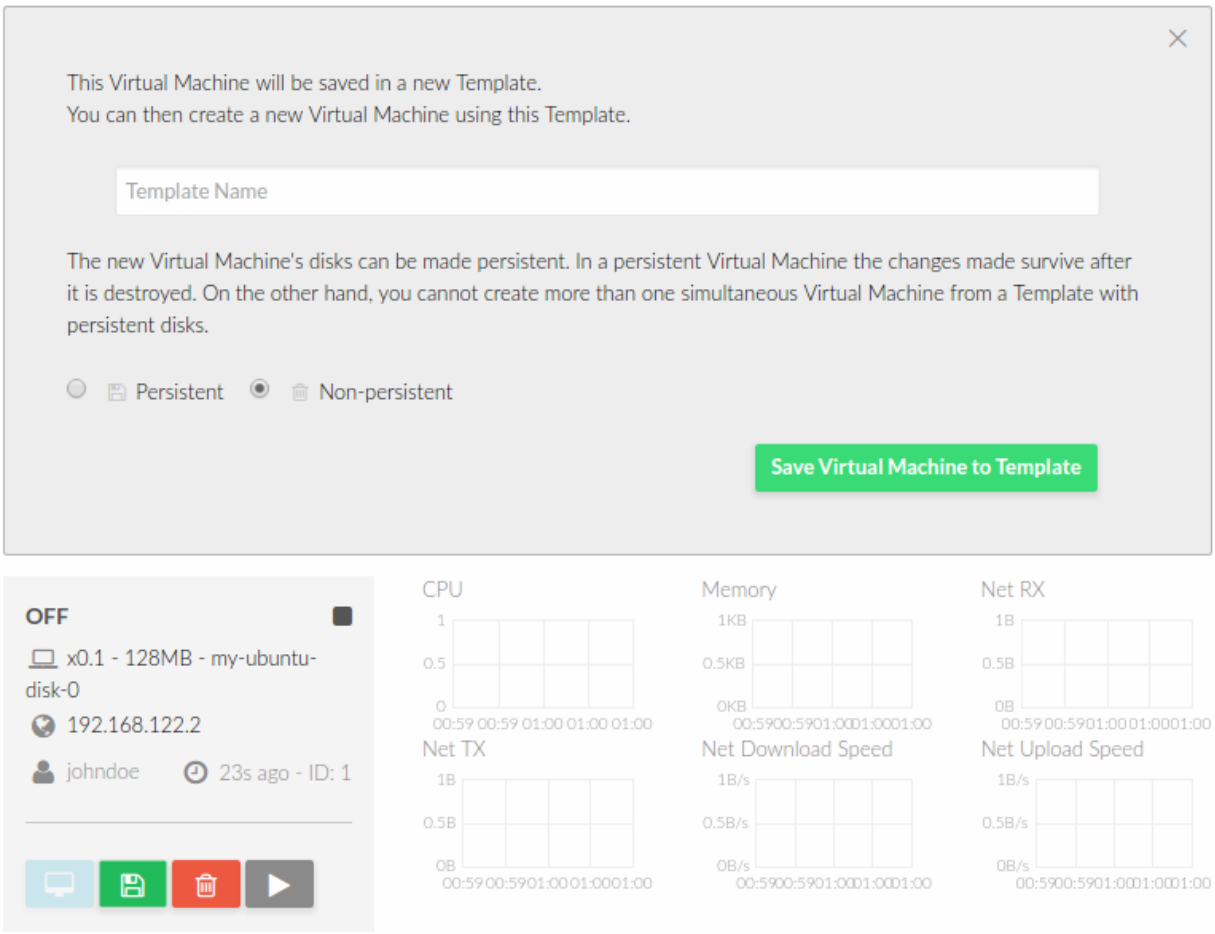
Make the VM Changes Persistent

Users can create a persistent private copy of the available templates. A persistent copy will preserve the changes made to the VM disks after the instance is terminated. This template is private, and will only be listed to the owner user.

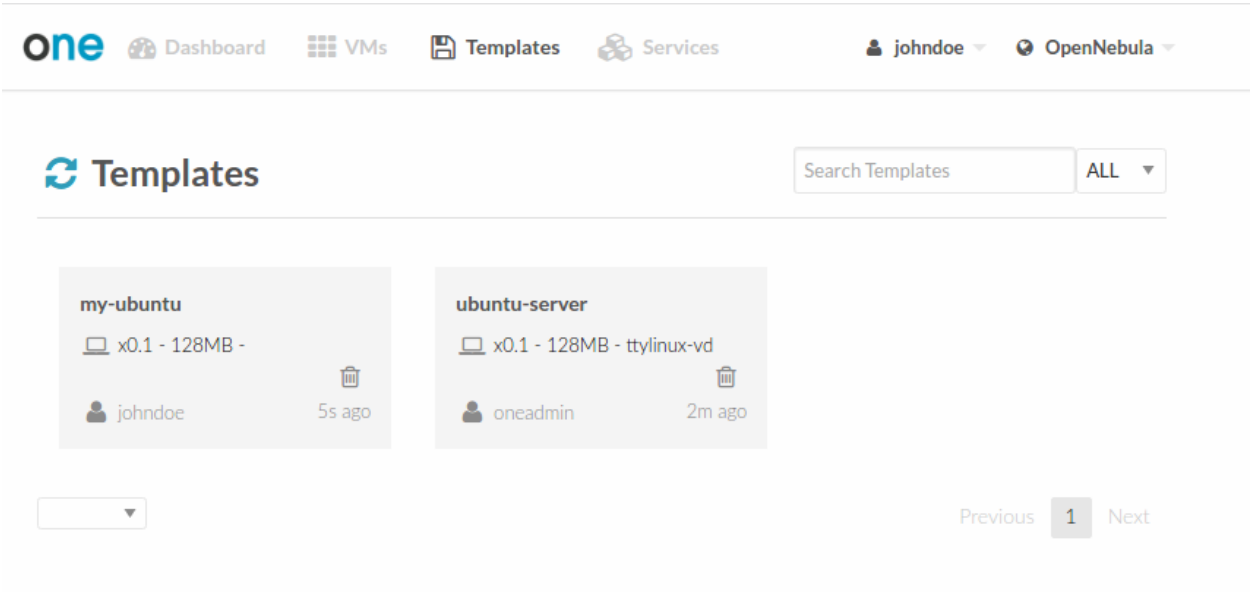
To create a persistent copy, use the “Persistent” switch next to the create button:



Alternatively, a VM that was not created as persistent can be saved before it is destroyed. To do so, the user has to power off the VM first and then use the save operation.



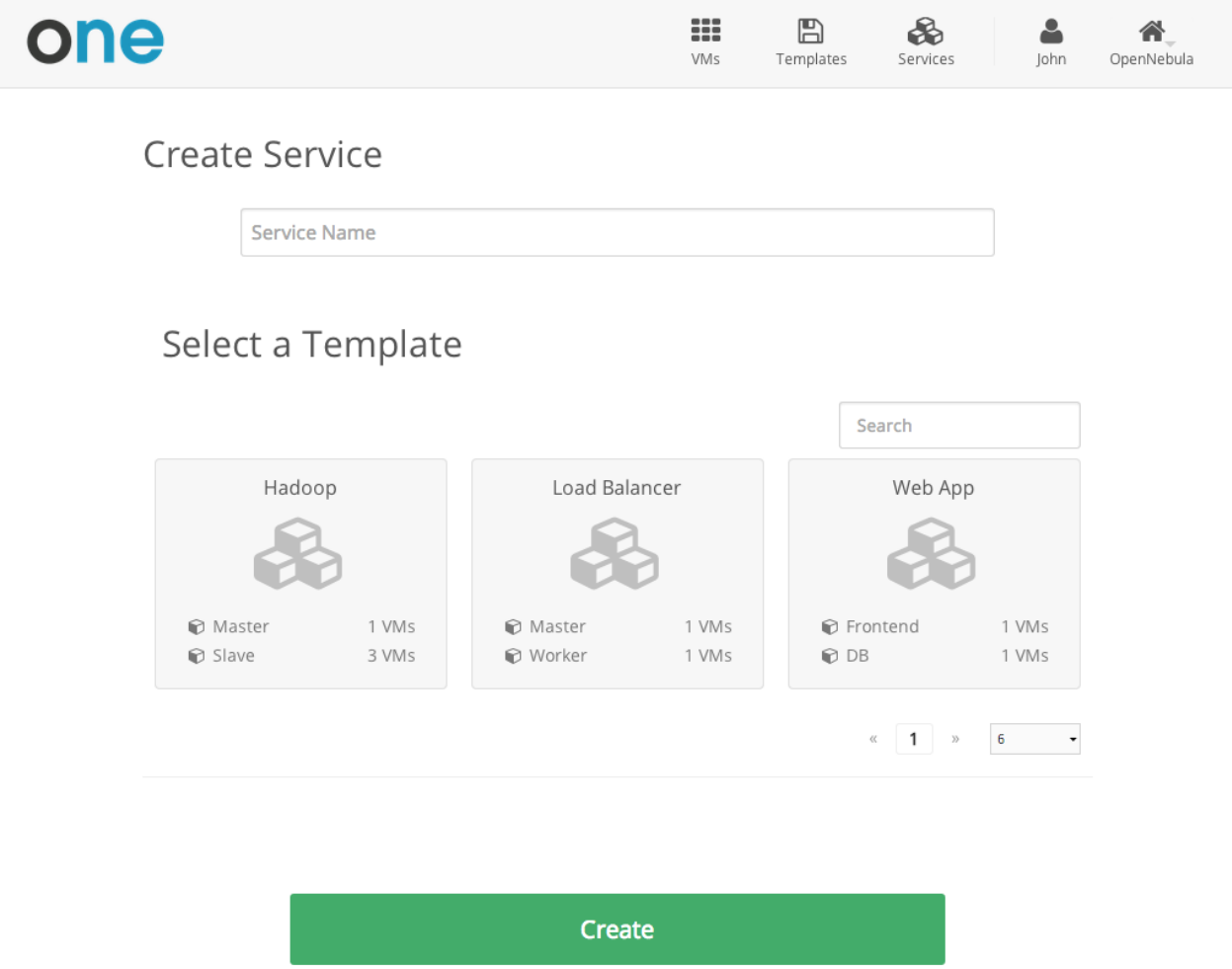
Any of the these two actions will create a new Template with the VM name. This template can be used in the "new VM wizard" to restore the VM after it is terminated. This template contains a copy of each one of the original disk images. If you delete this template, all the disk contents will be also lost.



Note: Avoid making a persistent copy of a persistent copy! Although there are use cases where it is justified, you will end with a long list of Templates and the disk usage quota will decrease quickly.

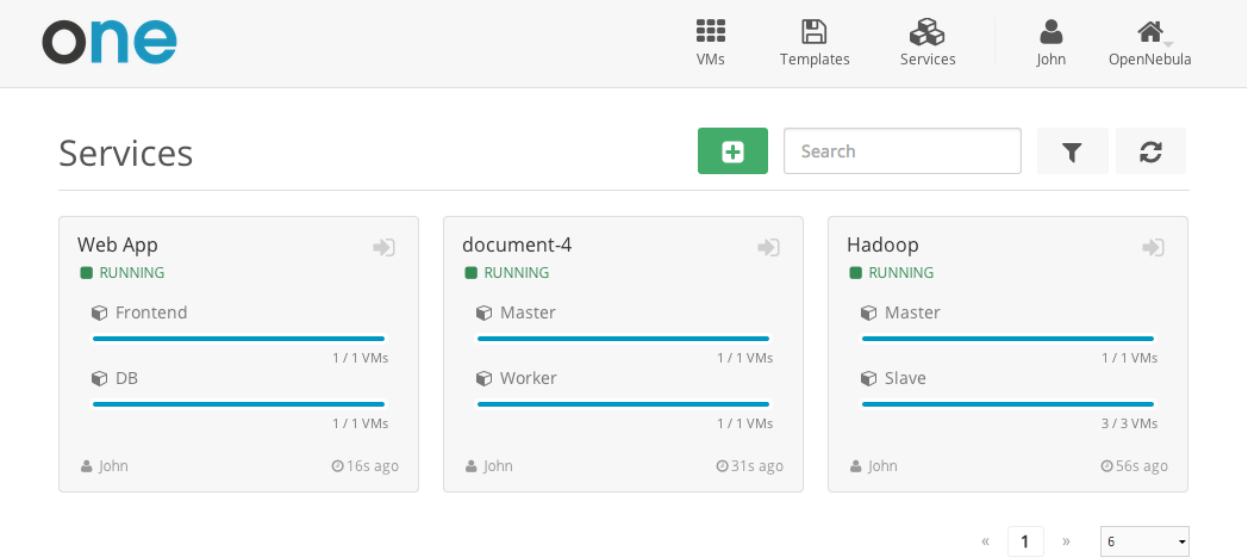
Create Service

In this scenario the cloud administrator must prepare a set of Service templates and make them available to the cloud users. These Service templates must be ready to be instantiated, i.e. they define all the mandatory attributes and the templates that are referenced are available for the user. Before using them, users can optionally customize the Service cardinality, define the network interfaces and provide values required by the template.

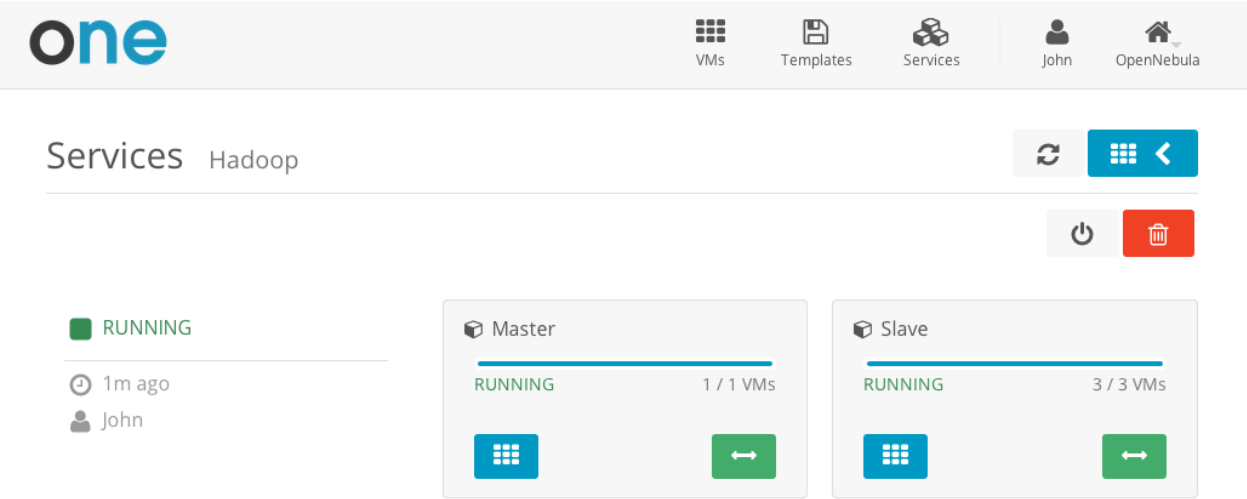


Manage Services

The status of the Services can be monitored from the Services tab



Information of the creation time, cardinality and status for each Role are available in the detailed view of the Service

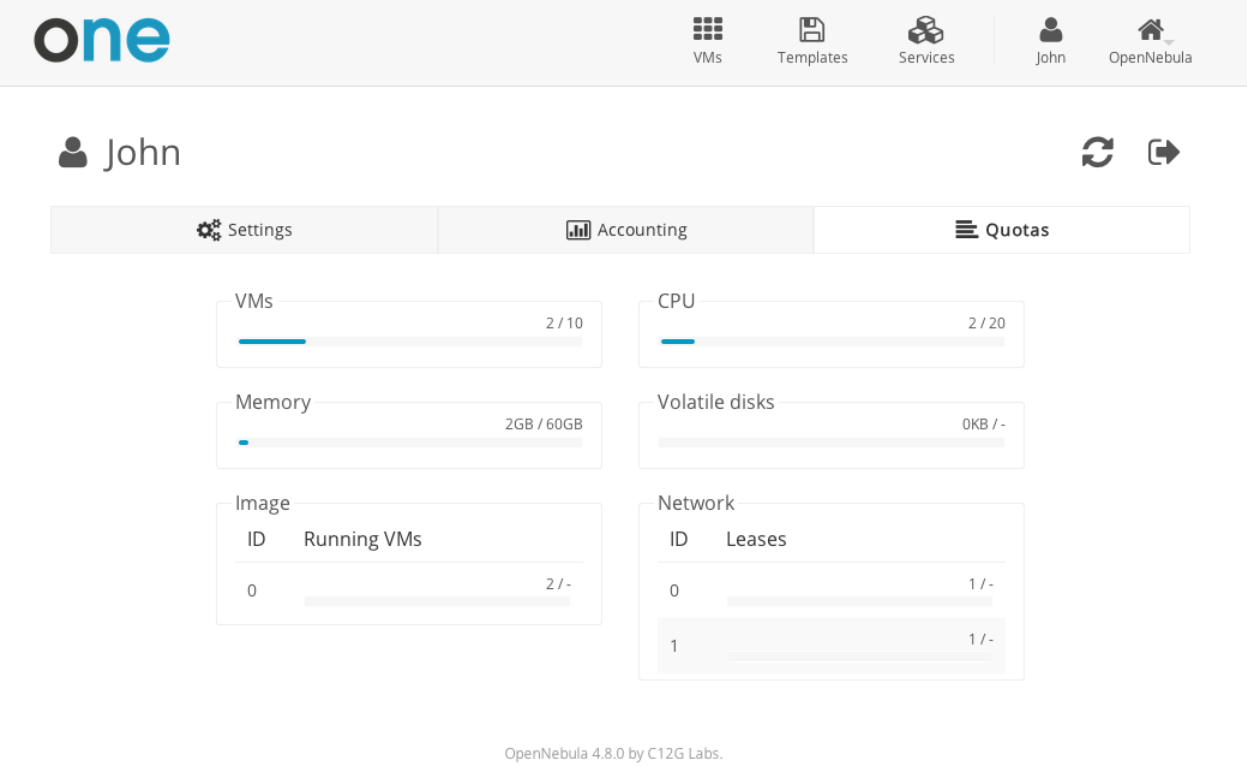


A user can perform the following actions from this view:

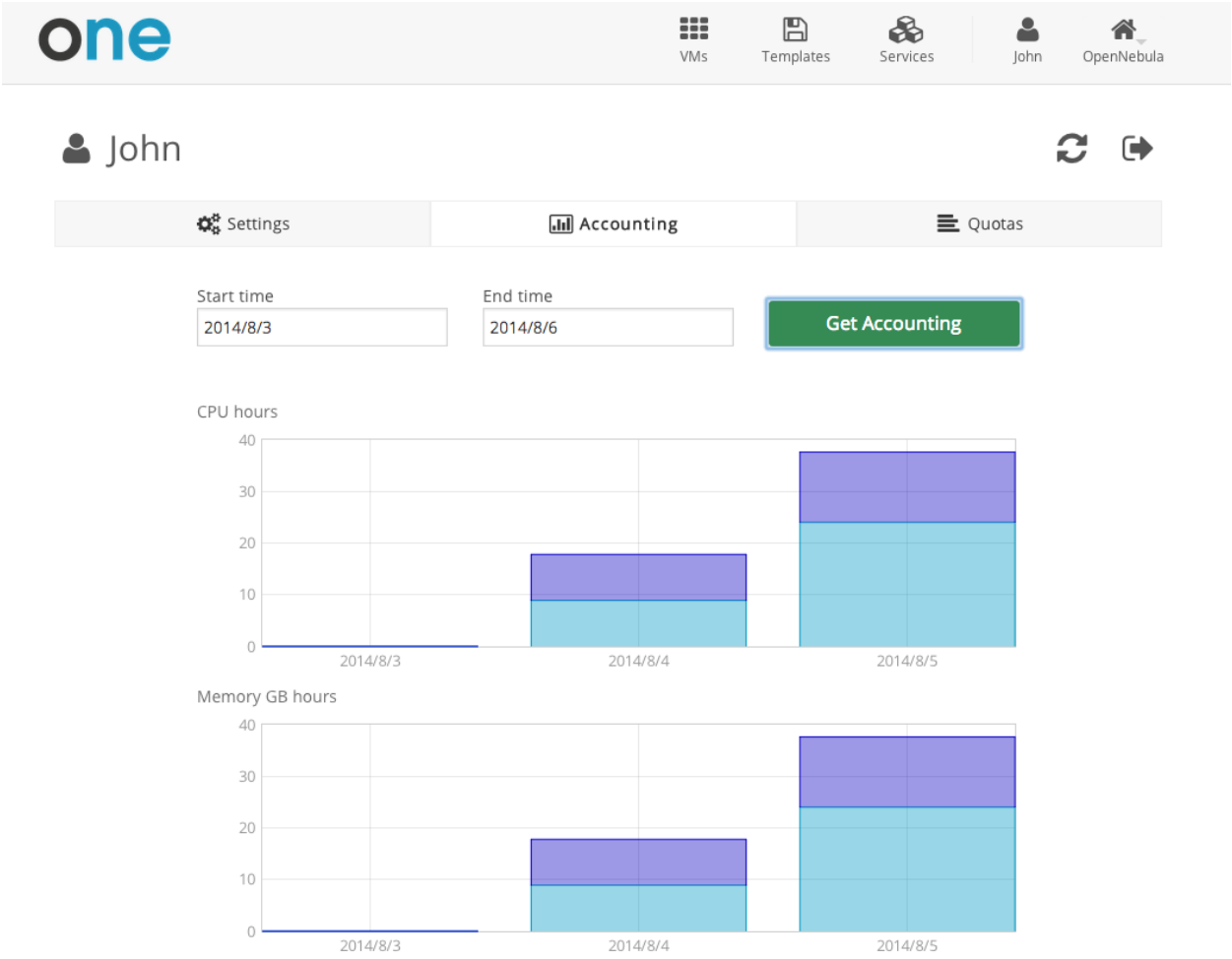
- Change the cardinality of each Role
- Retrieve the VMs of each Role
- Delete the Service
- Recover the Service from a fail status

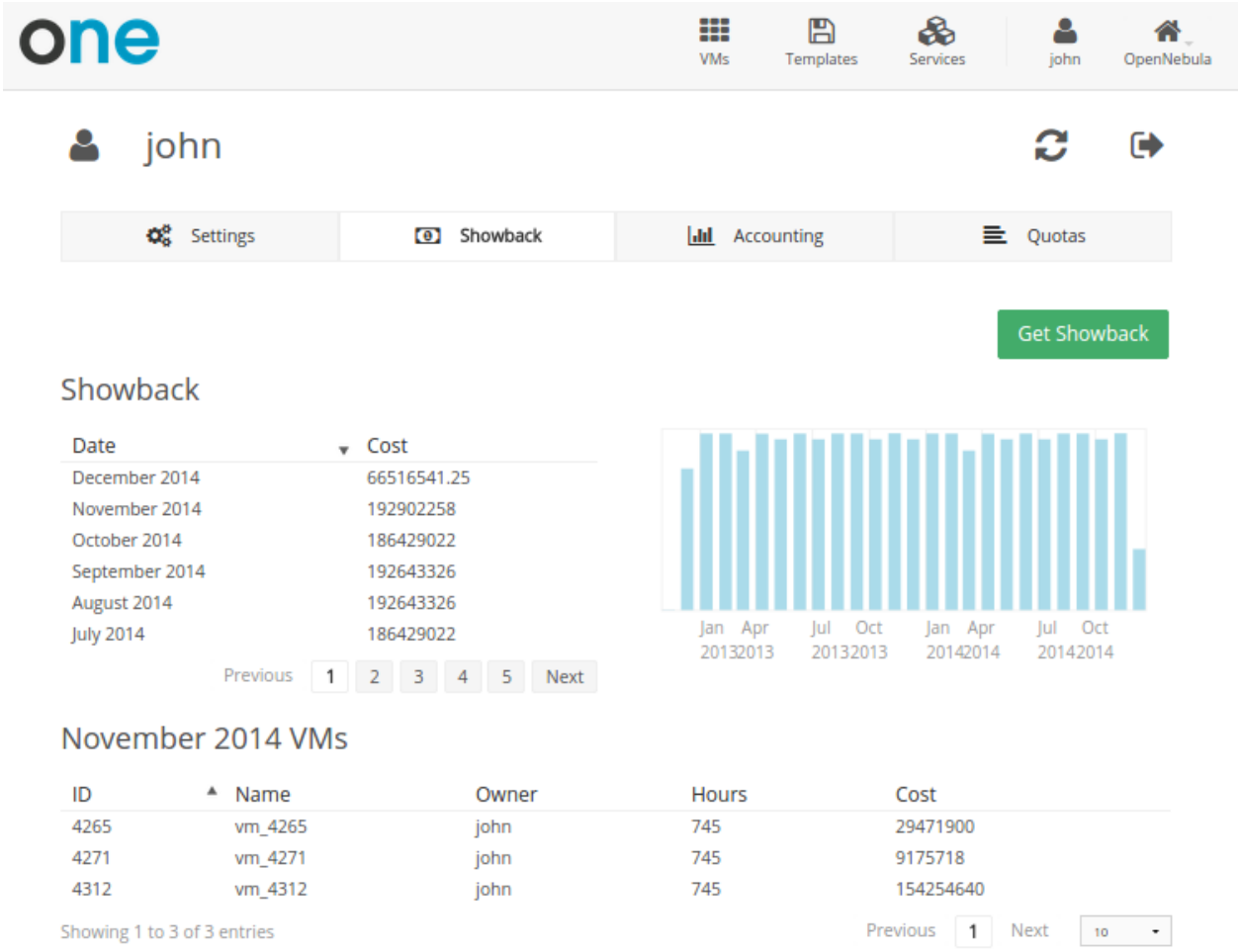
Usage, Accounting and Showback

The user can check his current usage and quotas



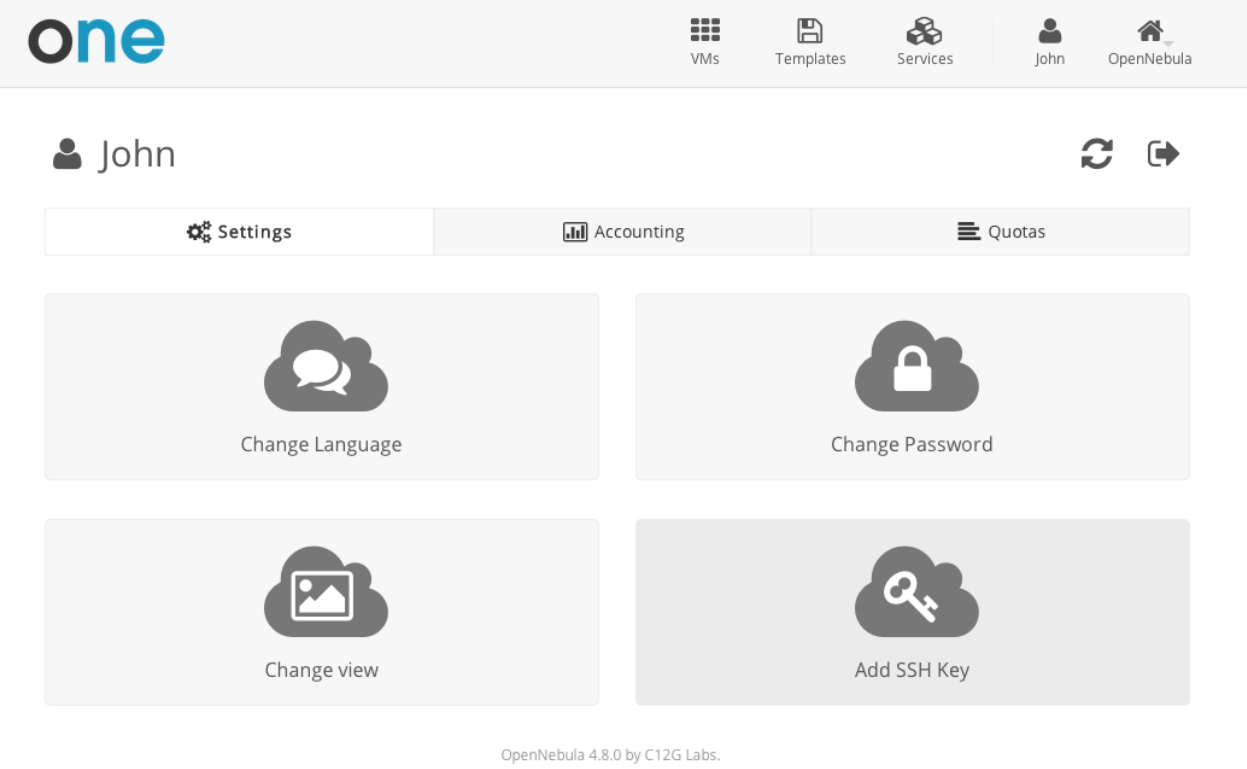
Also, the user can generate accounting reports for a given range of time


one
VMs
Templates
Services
John
OpenNebula
John
Settings
Accounting
Quotas
Start time
2014/8/3
End time
2014/8/6
Get Accounting
CPU hours
40
30
20
10
0
2014/8/3
2014/8/4
2014/8/5
Memory GB hours
40
30
20
10
0
2014/8/3
2014/8/4
2014/8/5



User Settings

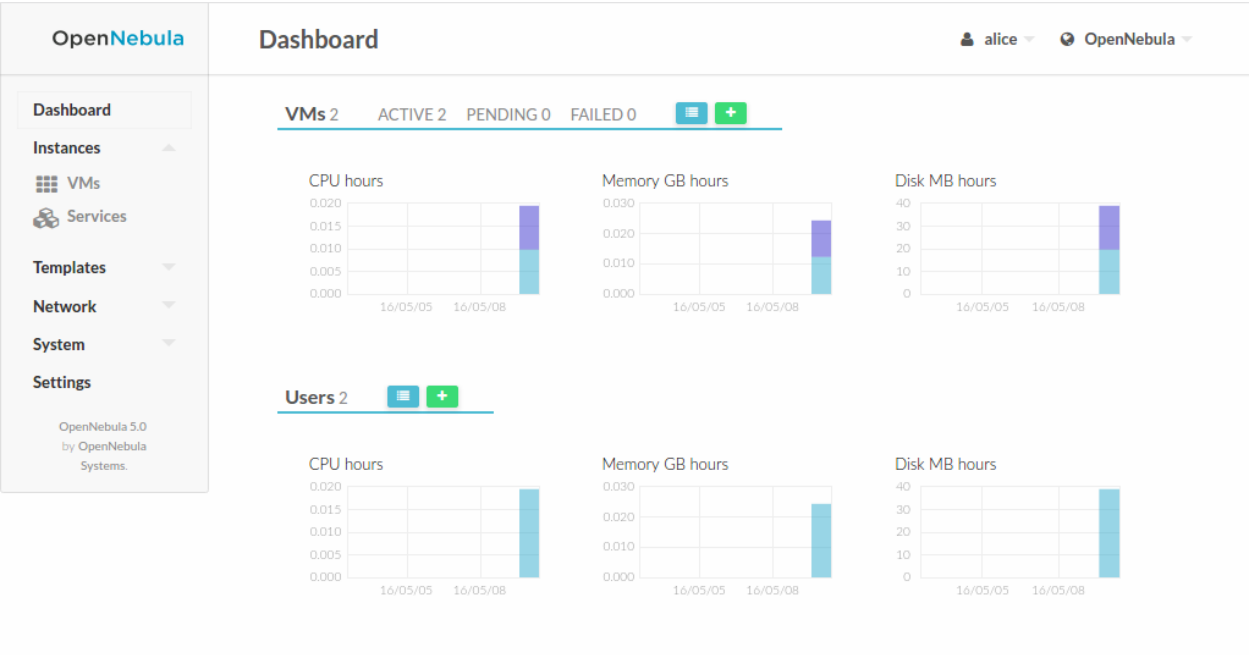
From the user settings tab, the user can change his password, language, ssh key and view

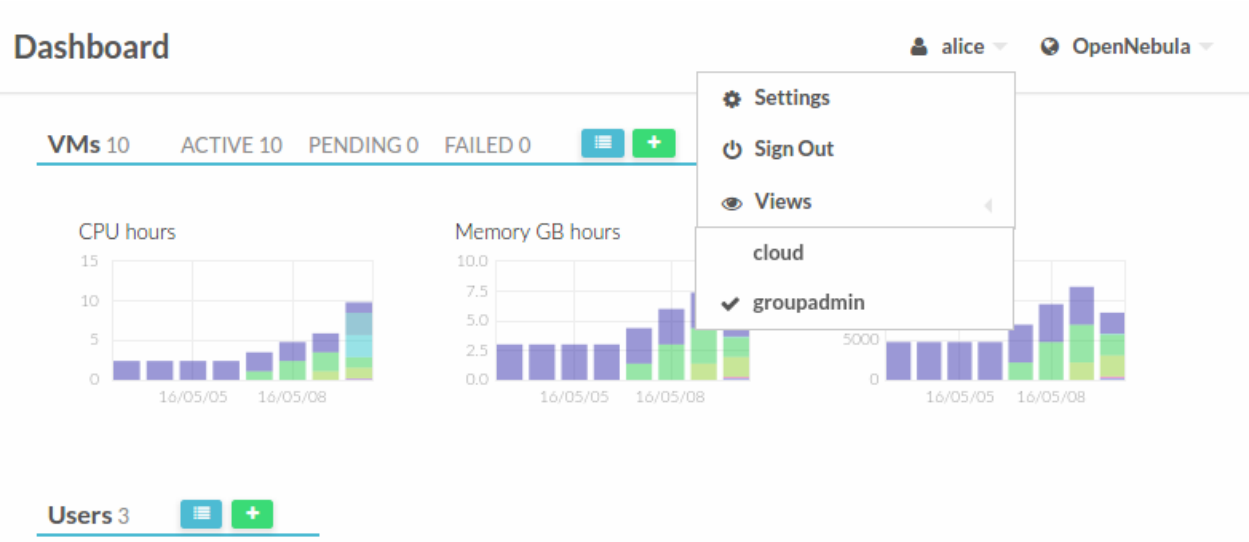


6.3 Group Admin View

The role of a Group Admin is to manage all the virtual resources of the Group, including the creation of new users. When one of these Group Admin users access vOnecloud, they get a limited version of the cloud administrator view. Groups can be confined to a subset of physical and virtual resources through *Virtual Data Centers*.

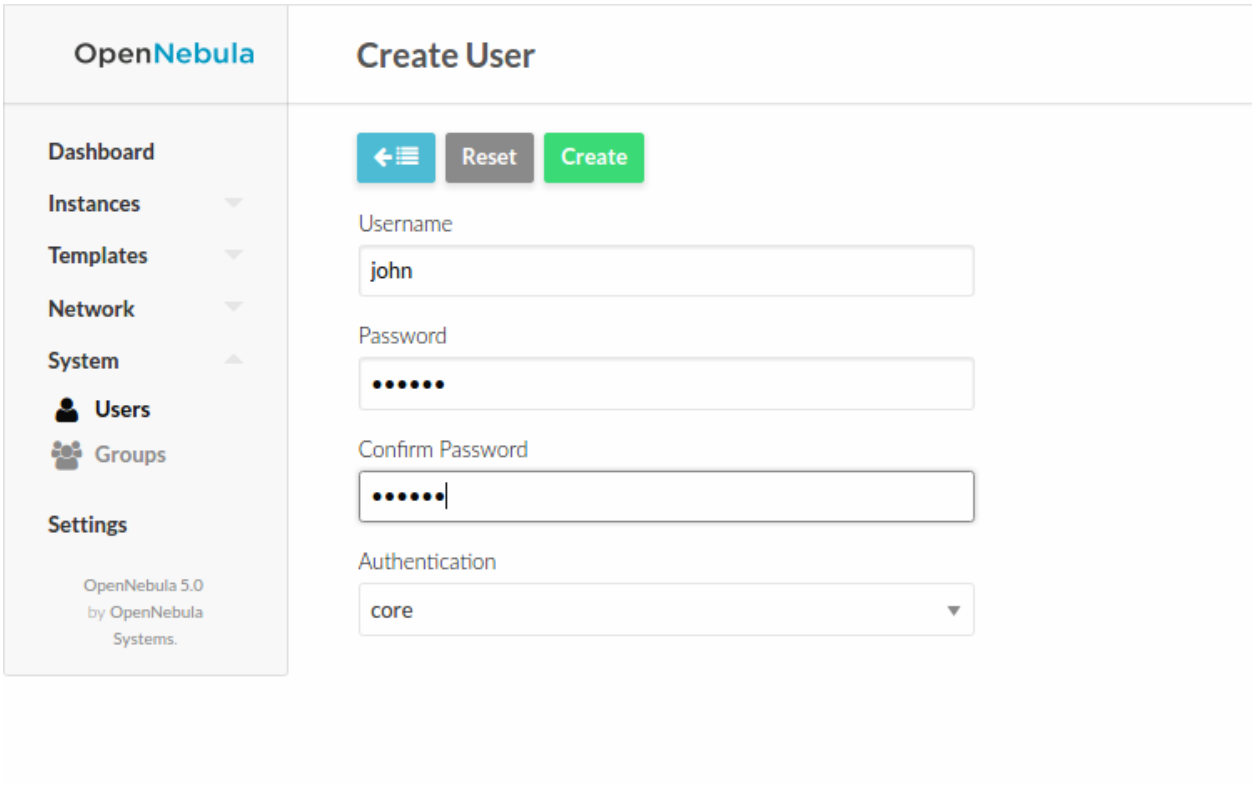
Group administrators can also access the *simplified Cloud View* if they prefer to.



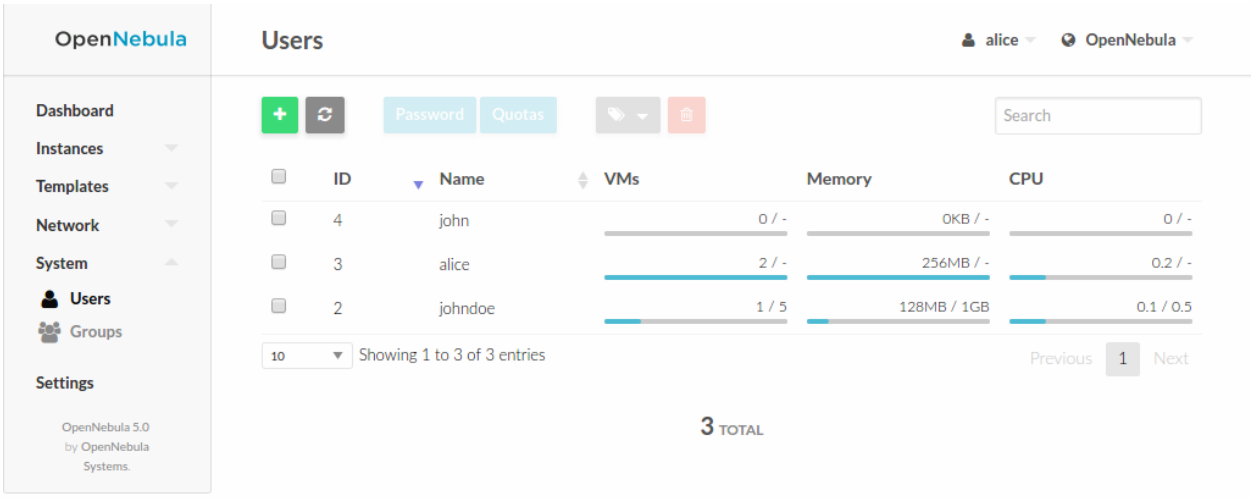


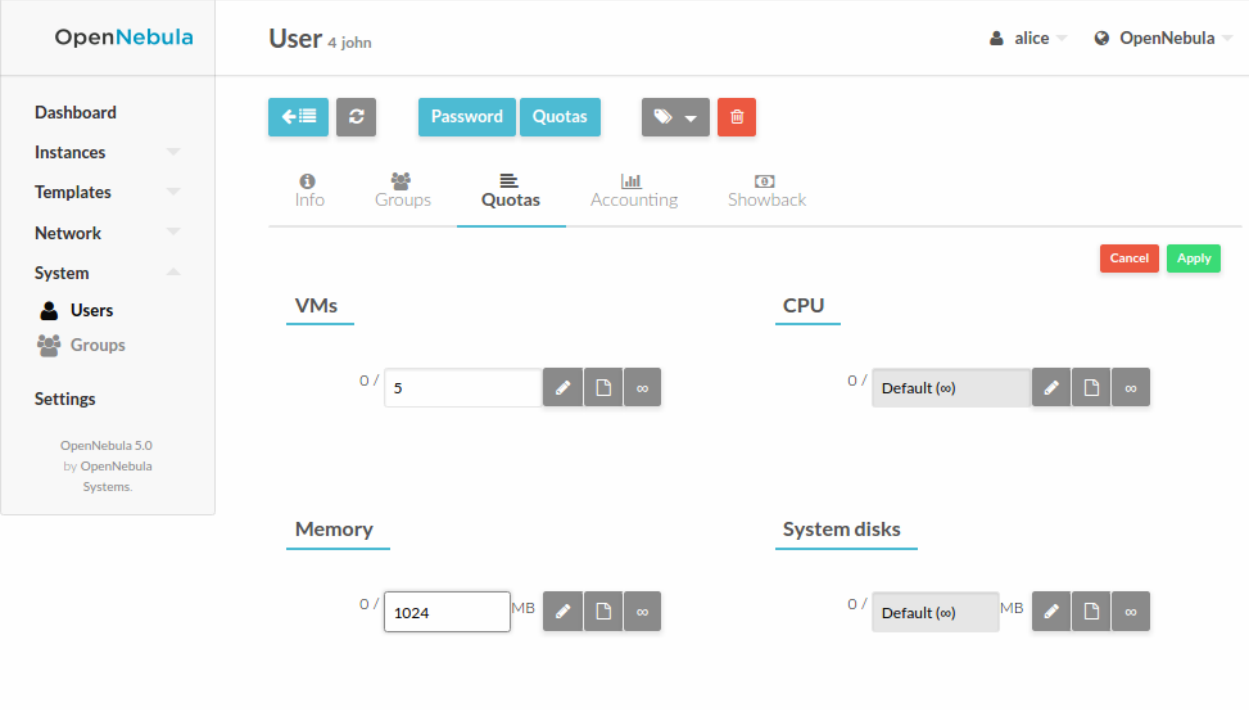
6.3.1 Manage Users

The Group Admin can create new user accounts, that will belong to the same Group.



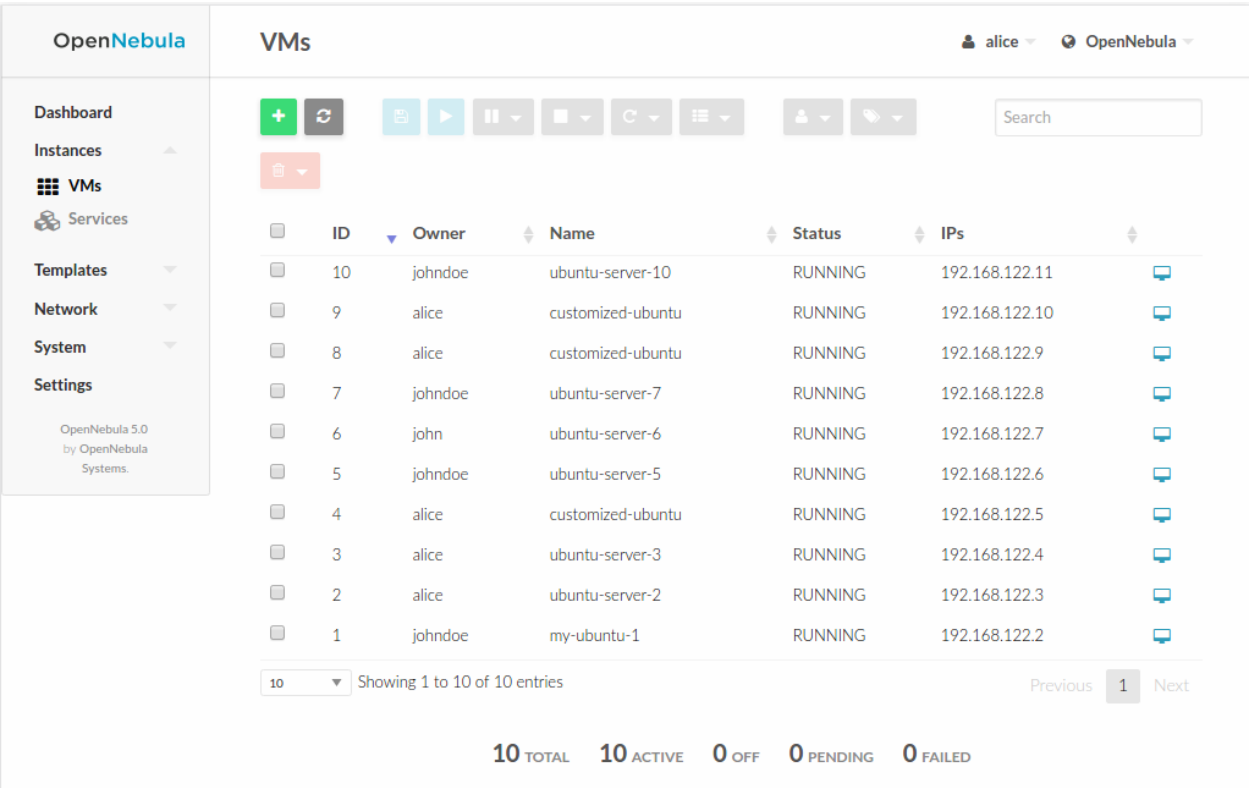
They can also see the current resource usage of all the Group users, and set quota limits for each one of them.





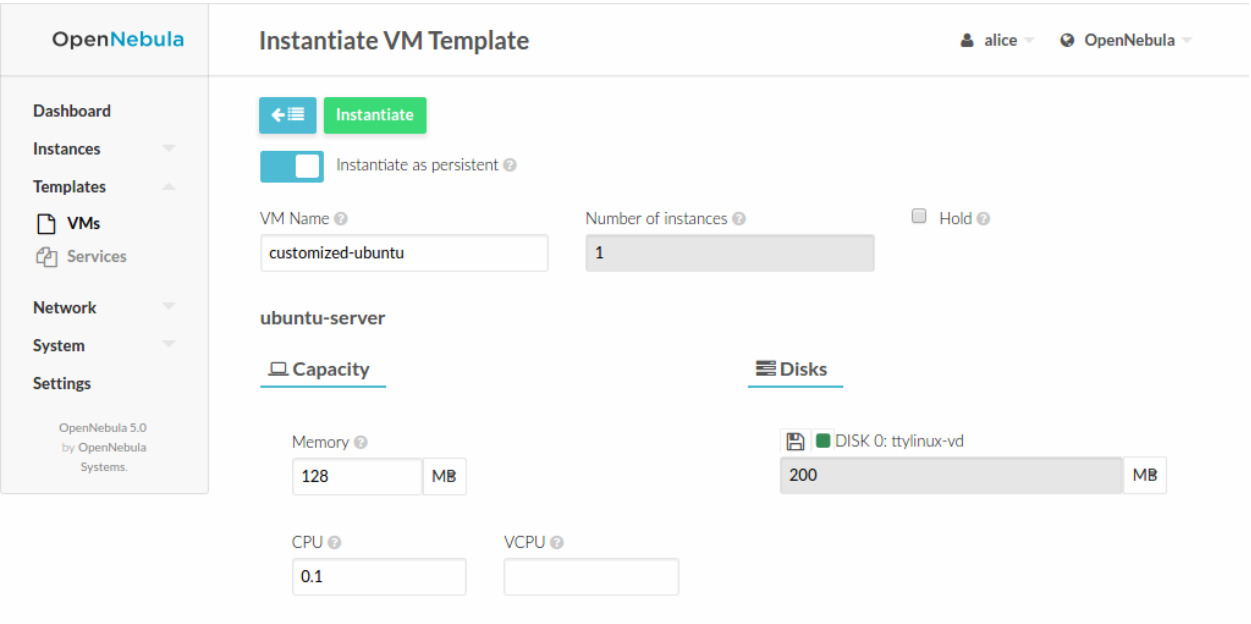
6.3.2 Manage Resources

The Group admin can manage the Services, VMs and Templates of other users in the Group.



6.3.3 Create Resources

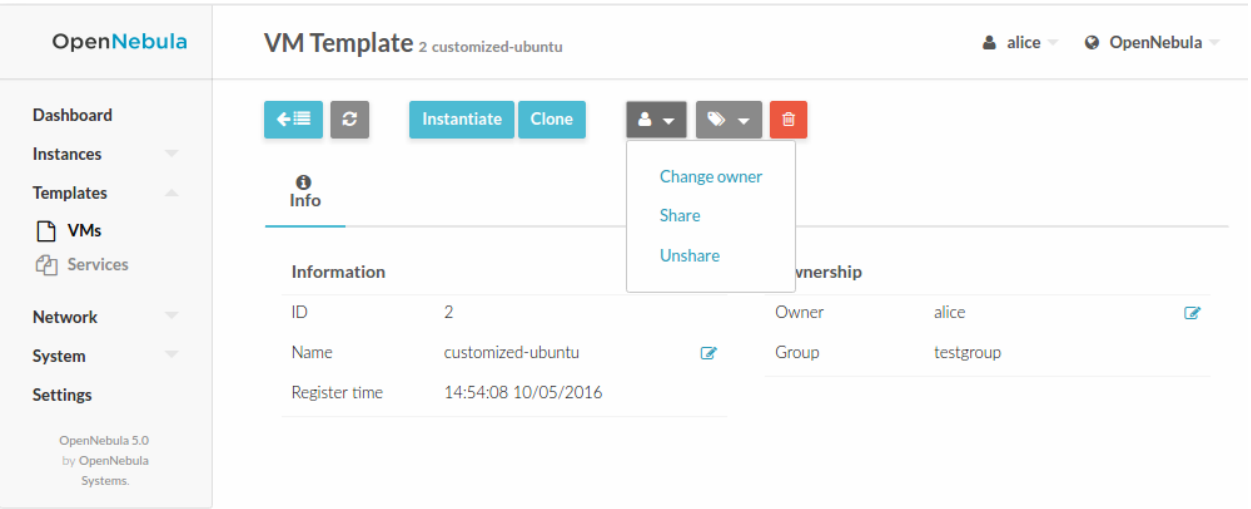
The Group admin can create new resources in the same way as a regular user does from the *Cloud view*. The creation wizard for the Virtual Machines and Services are similar in the `groupadmin` and `cloud` views.



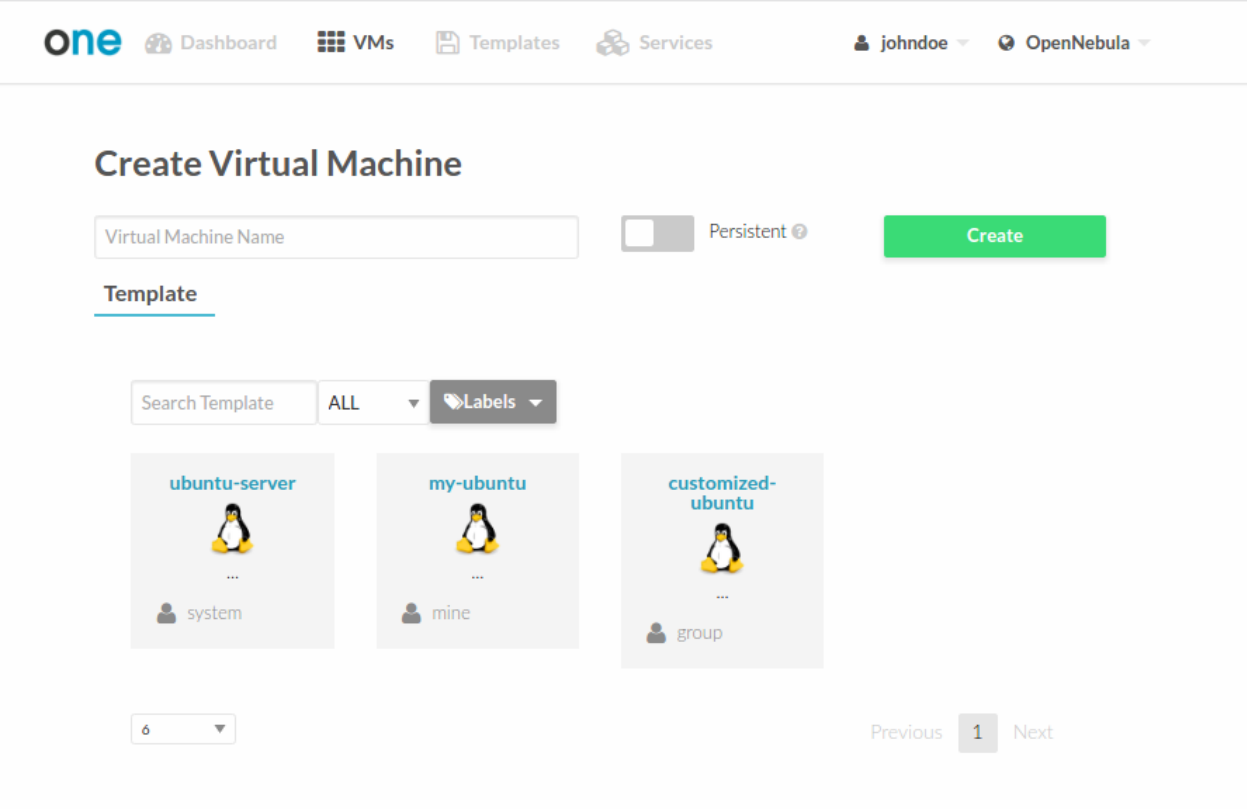
6.3.4 Prepare Resources for Other Users

Any user of the Cloud View or Group Admin View can save the changes made to a VM back to a new Template, and use this Template to instantiate new VMs later. See the *how to instantiate a VM to persistent* and *how to save as template* for more information.

The Group admin can also share his own Saved Templates with the rest of the group. For example the Group admin can instantiate a clean VM prepared by the cloud administrator, install software needed by other users in his Group, save it in a new Template and make it available for the rest of the group.



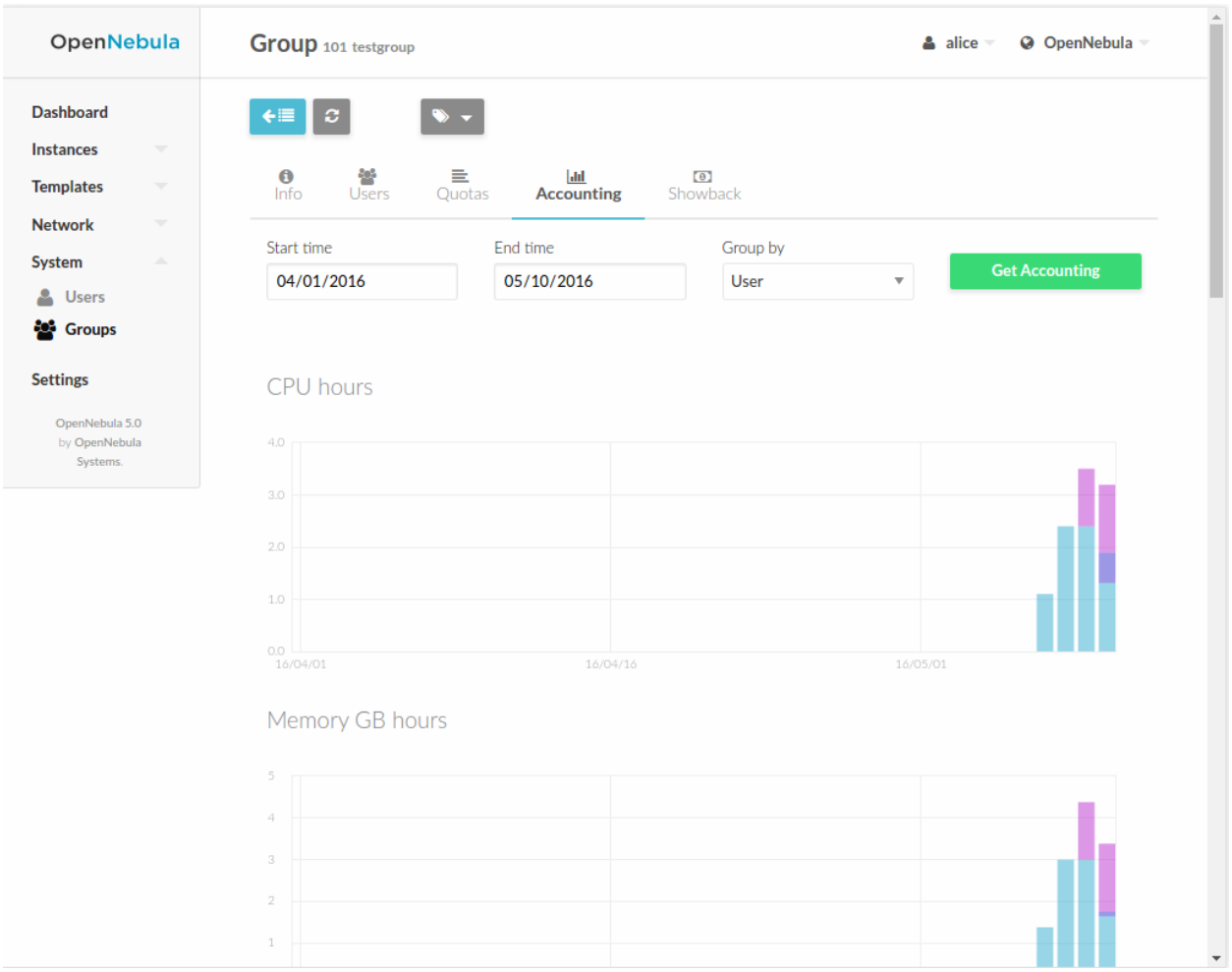
These shared templates will be listed to all the group users in the VM creation wizard, marked as 'group'. A Saved Template created by a regular user is only available for that user and is marked as 'mine'.

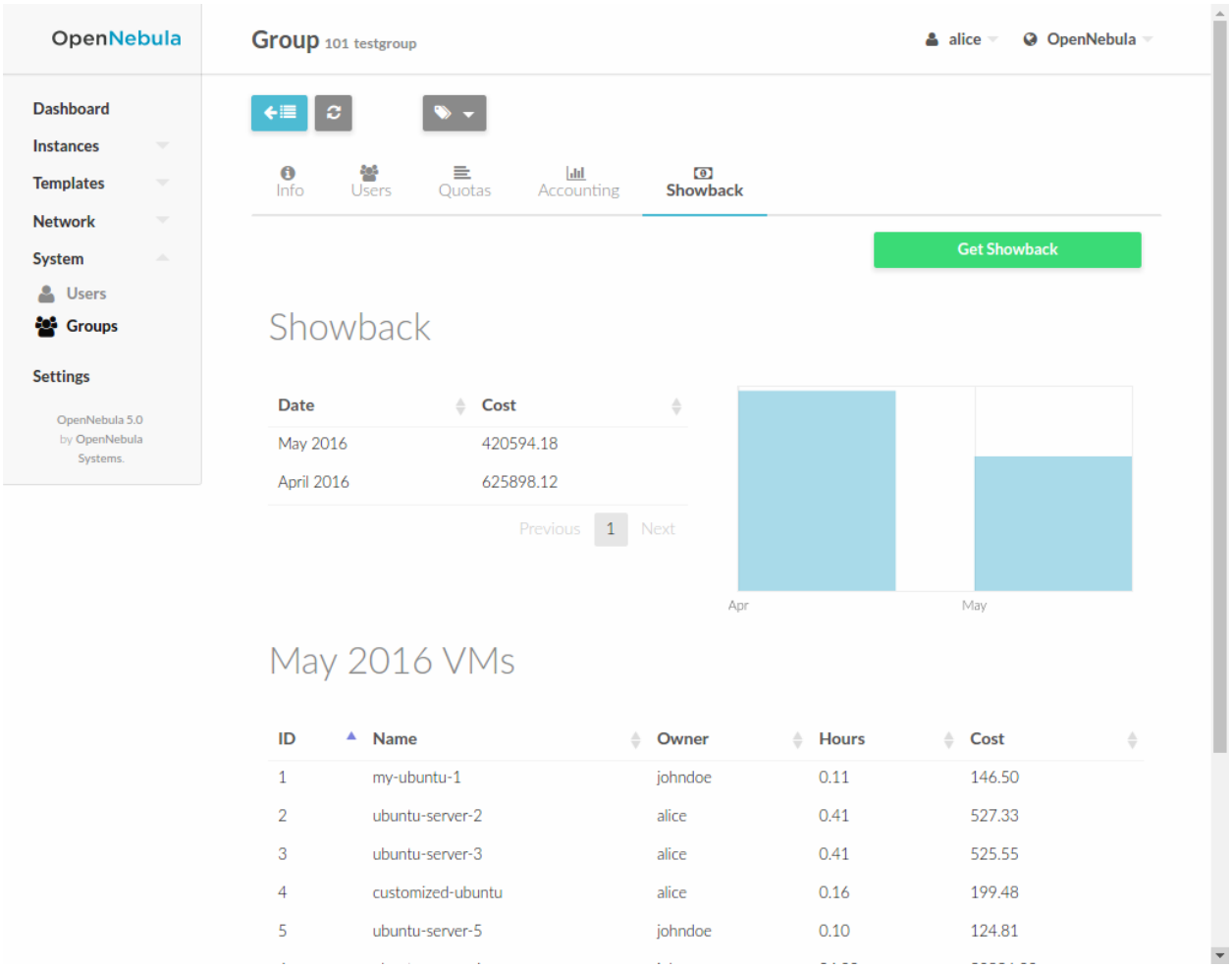


6.3.5 Accounting & Showback

Group Accounting & Showback

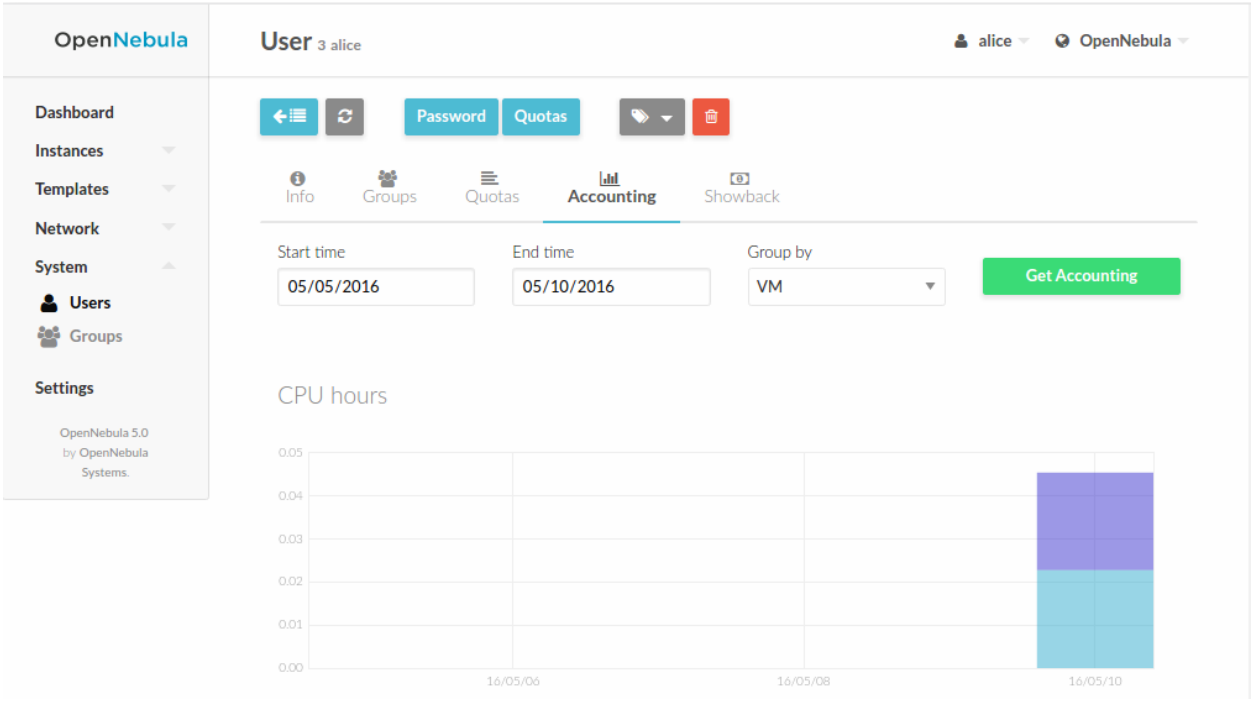
The Group info tab provides information of the usage of the Group and also accounting and showback reports can be generated. These reports can be configured to report the usage per VM or per user for a specific range of time.


OpenNebula
Group 101 testgroup
alice
OpenNebula
Dashboard
Instances
Templates
Network
System
Users
Groups
Settings
OpenNebula 5.0
by OpenNebula
Systems.
Info
Users
Quotas
Accounting
Showback
Start time
04/01/2016
End time
05/10/2016
Group by
User
Get Accounting
CPU hours
4.0
3.0
2.0
1.0
0.0
16/04/01
16/04/16
16/05/01
Memory GB hours
5
4
3
2
1



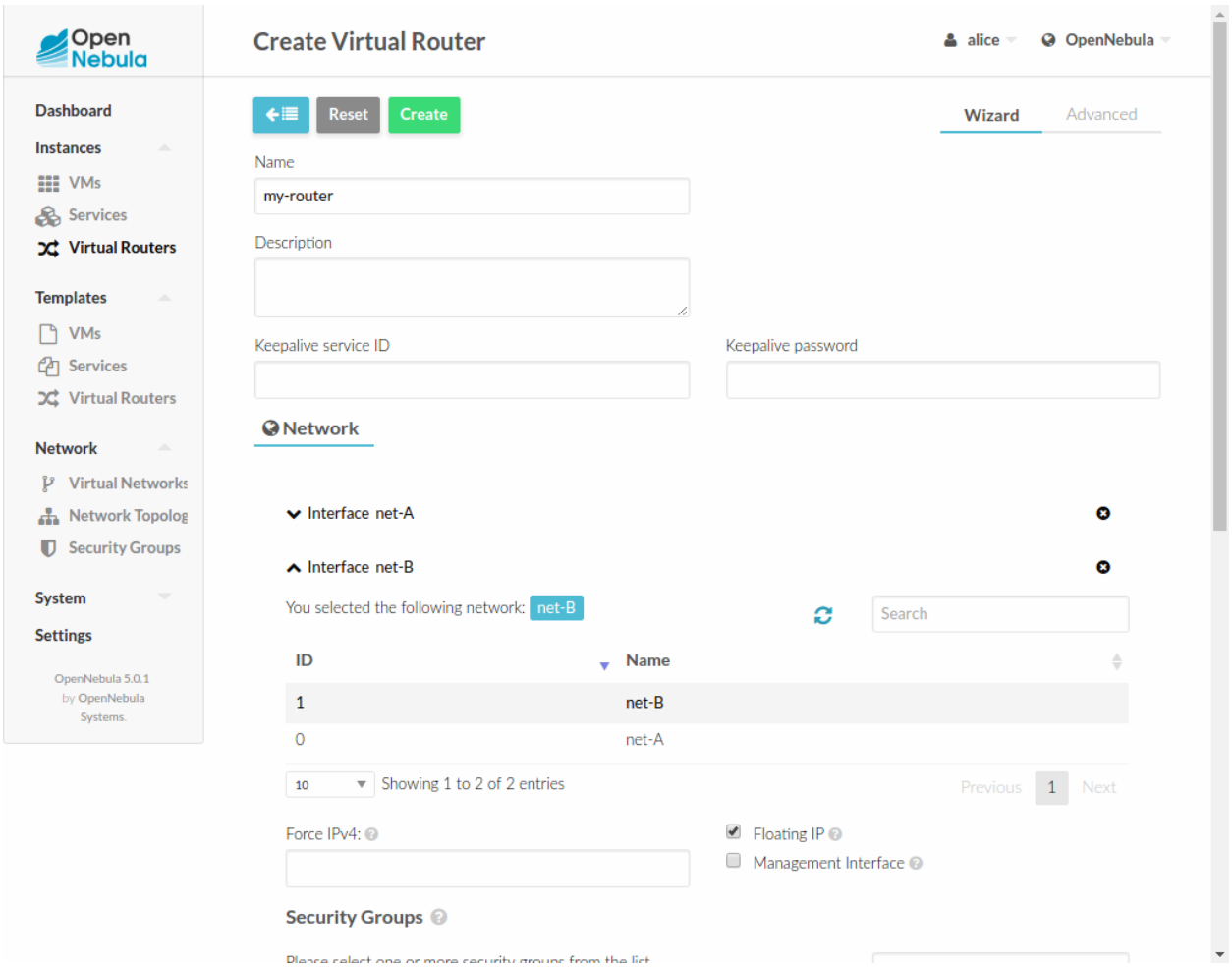
User Accounting & Showback

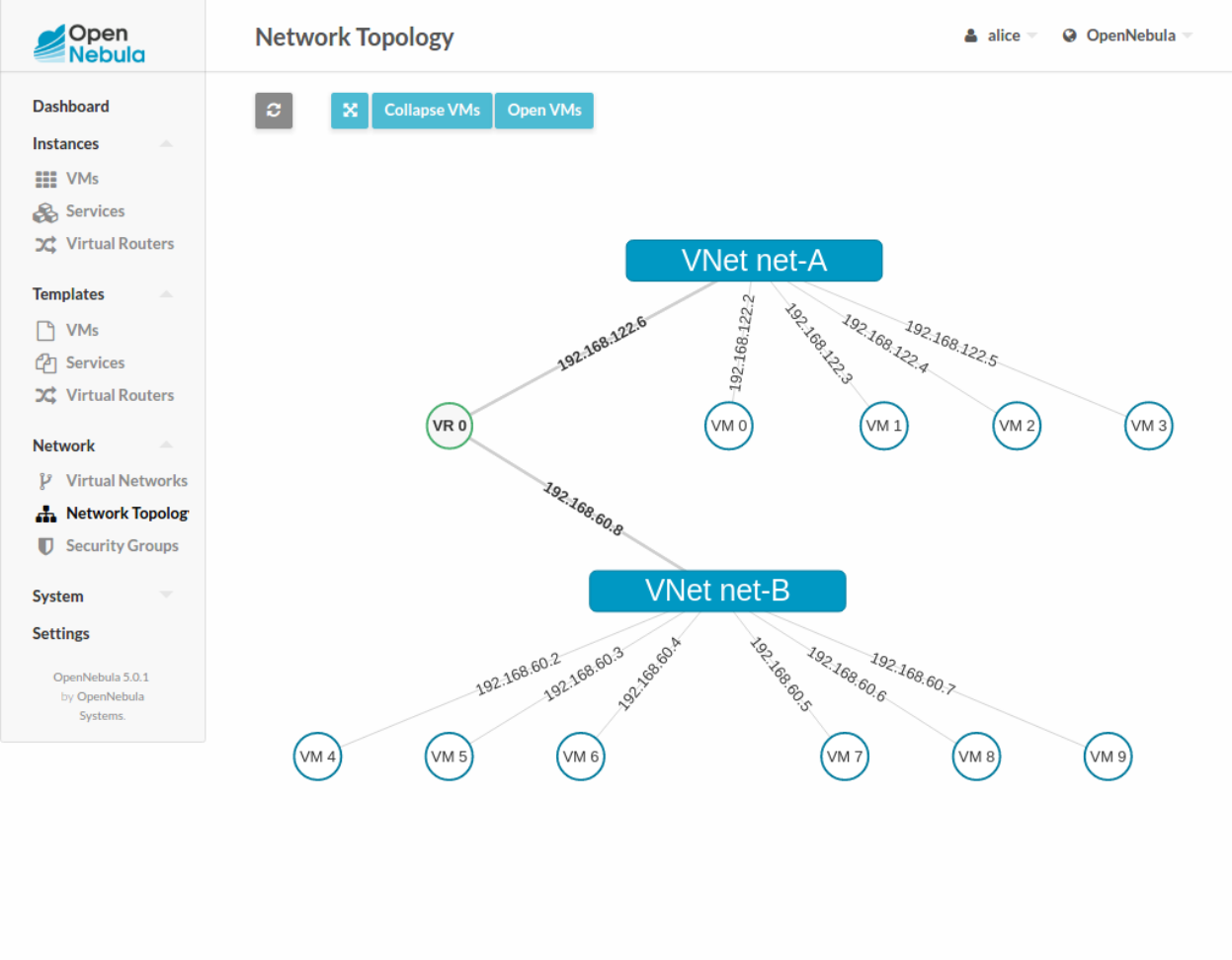
The detailed view of the user provides information of the usage of the user, from this view accounting reports can be also generated for this specific user



6.3.6 Networking

Group administrators can create *Virtual Routers* from Templates prepared by the cloud administrator. These Virtual Routers can be used to connect two or more of the Virtual Networks assigned to the Group.


Open Nebula
Create Virtual Router
alice
OpenNebula
Dashboard
Instances
VMs
Services
Virtual Routers
Templates
VMs
Services
Virtual Routers
Network
Virtual Networks
Network Topolog
Security Groups
System
Settings
OpenNebula 5.0.1 by OpenNebula Systems.
Reset
Create
Wizard
Advanced
Name
my-router
Description
Keepalive service ID
Keepalive password
Network
Interface net-A
Interface net-B
You selected the following network: net-B
Search
ID
Name
1
net-B
0
net-A
10
Showing 1 to 2 of 2 entries
Previous
1
Next
Force IPv4:
Floating IP
Management Interface
Security Groups


Open Nebula
Network Topology
alice
OpenNebula
Dashboard
Instances
VMs
Services
Virtual Routers
Templates
VMs
Services
Virtual Routers
Network
Virtual Networks
Network Topolog
Security Groups
System
Settings
OpenNebula 5.0.1
by OpenNebula
Systems.
Collapse VMs
Open VMs
VNet net-A
192.168.122.6
192.168.122.2
192.168.122.3
192.168.122.4
192.168.122.5
VR 0
VM 0
VM 1
VM 2
VM 3
192.168.60.8
VNet net-B
192.168.60.2
192.168.60.3
192.168.60.4
192.168.60.5
192.168.60.6
192.168.60.7
VM 4
VM 5
VM 6
VM 7
VM 8
VM 9

CHAPTER

SEVEN

INFRASTRUCTURE CONFIGURATION

7.1 Introduction

Now that you are familiar with vOneCloud concepts and operations, it is time to extend its functionality by adding new infrastructure components and reviewing advanced configurations and operations.

7.2 Add New vCenter Resources and Advanced Features

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.

It is important to note that there are different behavior of the vCenter resources when deleted in OpenNebula. The following resources are NOT deleted in vCenter when deleted in OpenNebula:

- VM Templates
- Datastores

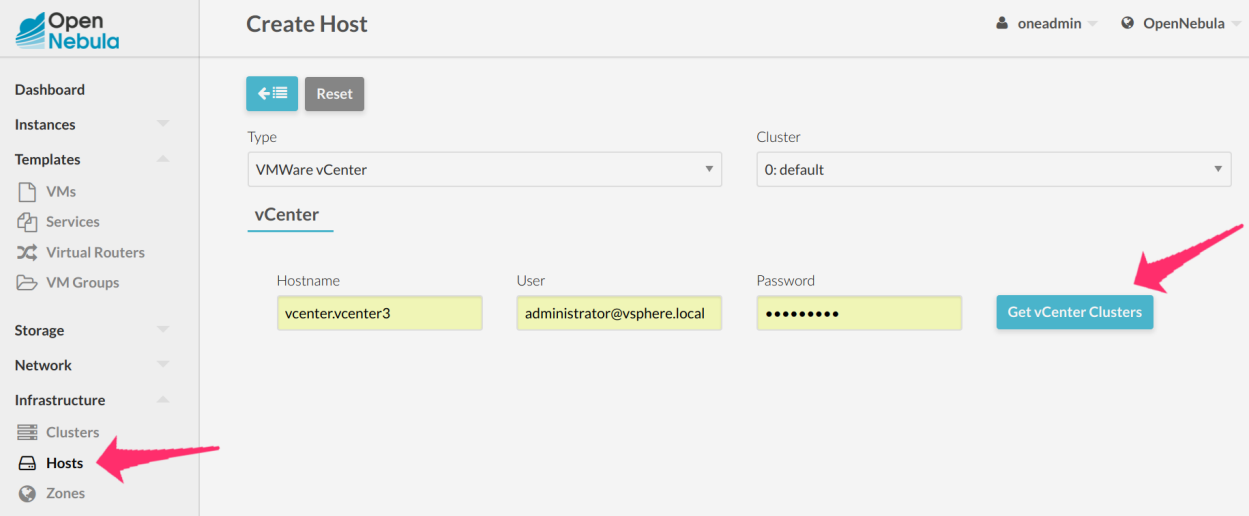
The following resource are deleted in vCenter when deleted in OpenNebula:

- Images
- Virtual Machines
- Networks

If resources are imported in vOneCloud rather than created throught it, they **won't** be deleted in vCenter if deleted in vOneCloud.

7.2.1 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.

7.2.2 Add New / Update VM Template

Let's see an example:

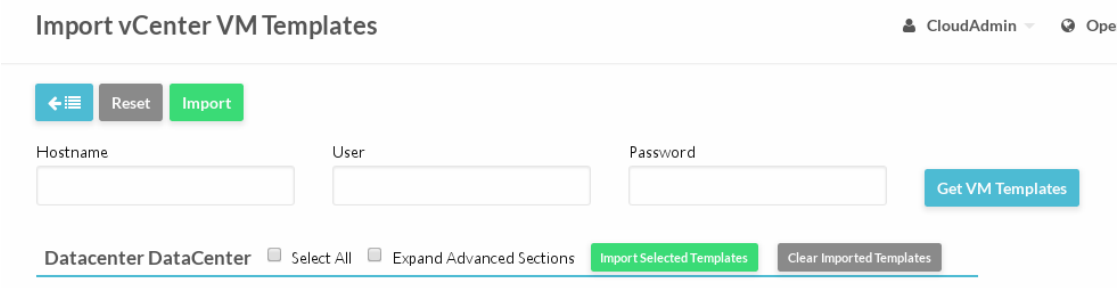
To avoid misunderstandings, there are two VM templates we will refer to: the vOneCloud VM Templates and the vCenter VM Templates. The formers are created in the vOneCloud web interface (Sunstone), whereas the latter are created directly through the vCenter Web Client.

A cloud administrator builds two vOneCloud templates to represent one vCenter VM Template available 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, you have to choices:

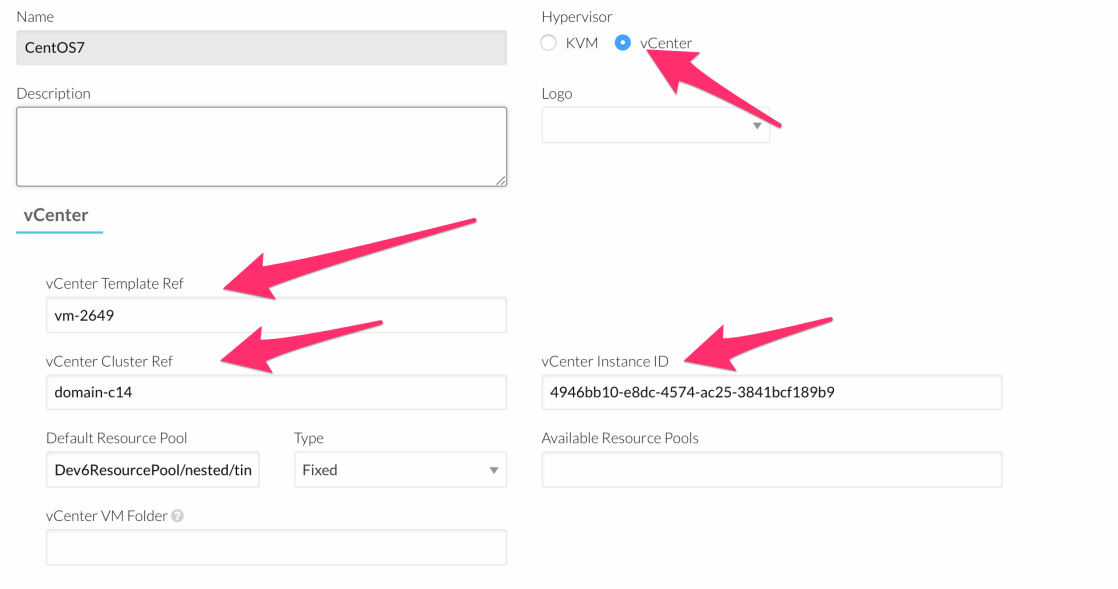
- Import an existing vCenter VM Template
- Create a new VM Template manually providing a reference to a vCenter VM Template

You can import a vCloud VM Template using the Import button and providing your vCenter's hostname or IP address and the credentials for a user with the required privileges. Then, select the templates you want to import and click the Import Selected Templates button.



If you want to create the vOneCloud VM Template manually, log in into Sunstone as **CloudAdmin** user as in explained *here*, proceed to the `Templates -> VMs`, and click on the **+** sign. Select *vCenter* as the hypervisor, and type in the *vCenter Instance ID*, the *vCenter Templated Ref* and the *vCenter Cluster Ref* (more information here). 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



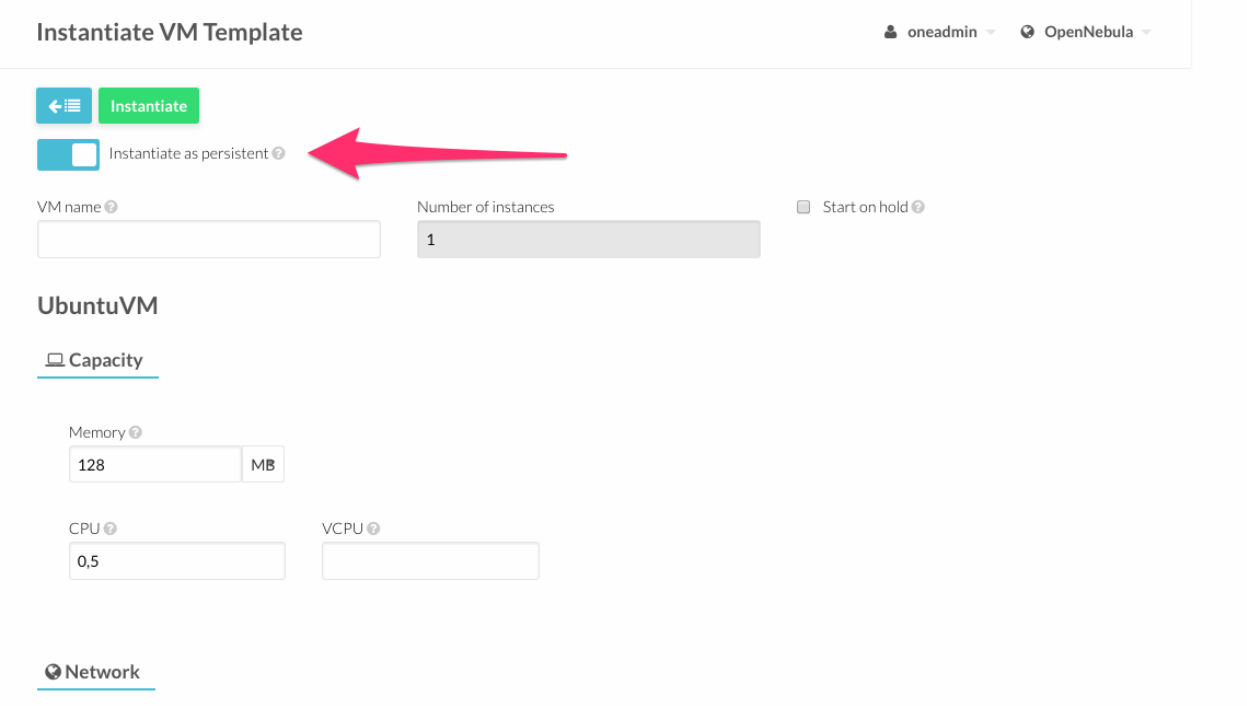
Create one VM Template with informaton about host vCenterA. Repeat for vCenterB.

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

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

The following advanced features can be used in the VM Templates and VMs. **VM Instantiate to Persistent**

At the time of deploying a VM Template, a flag can be used to create a new VM Template out of the VM.



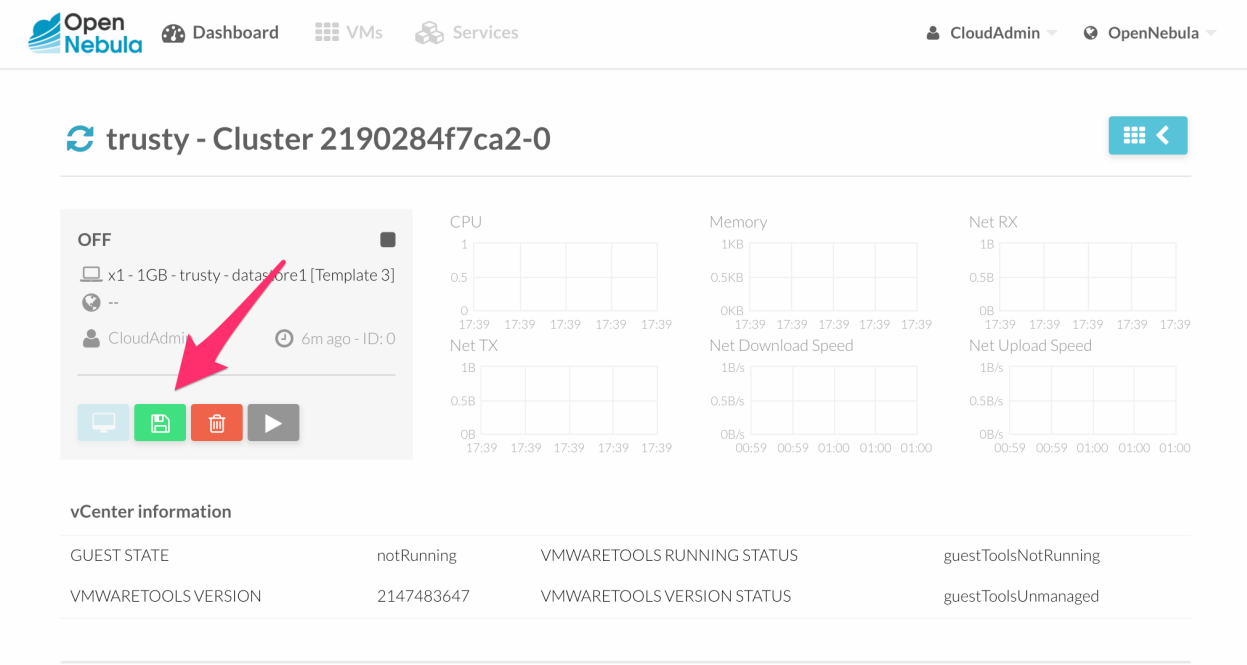
Whenever the VM life-cycle ends, vOneCloud will instruct vCenter to create a new vCenter VM Template out of the VM, with the settings of the VM including any new disks or network interfaces added through vOneCloud. Any new disk added to the VM will be saved as part of the template, and when a new VM is spawned from this new VM Template the disk will be cloned by vOneCloud (ie, it will no longer be persistent).

This functionality is very useful to create new VM Templates from a original VM Template, changing the VM configuration and/or installing new software, to create a complete VM Template catalog.

Note: A new vOneCloud VM Template will be created pointing to this new VM Template, so it can be instantiated through vOneCloud. This new vOneCloud VM Template will be pointing to the original template until the VM is shutdown, at which point it will be converted to a vCenter VM Template and the vOneCloud VM Template updated to point to this new vCentre VM Template.

Save as Template

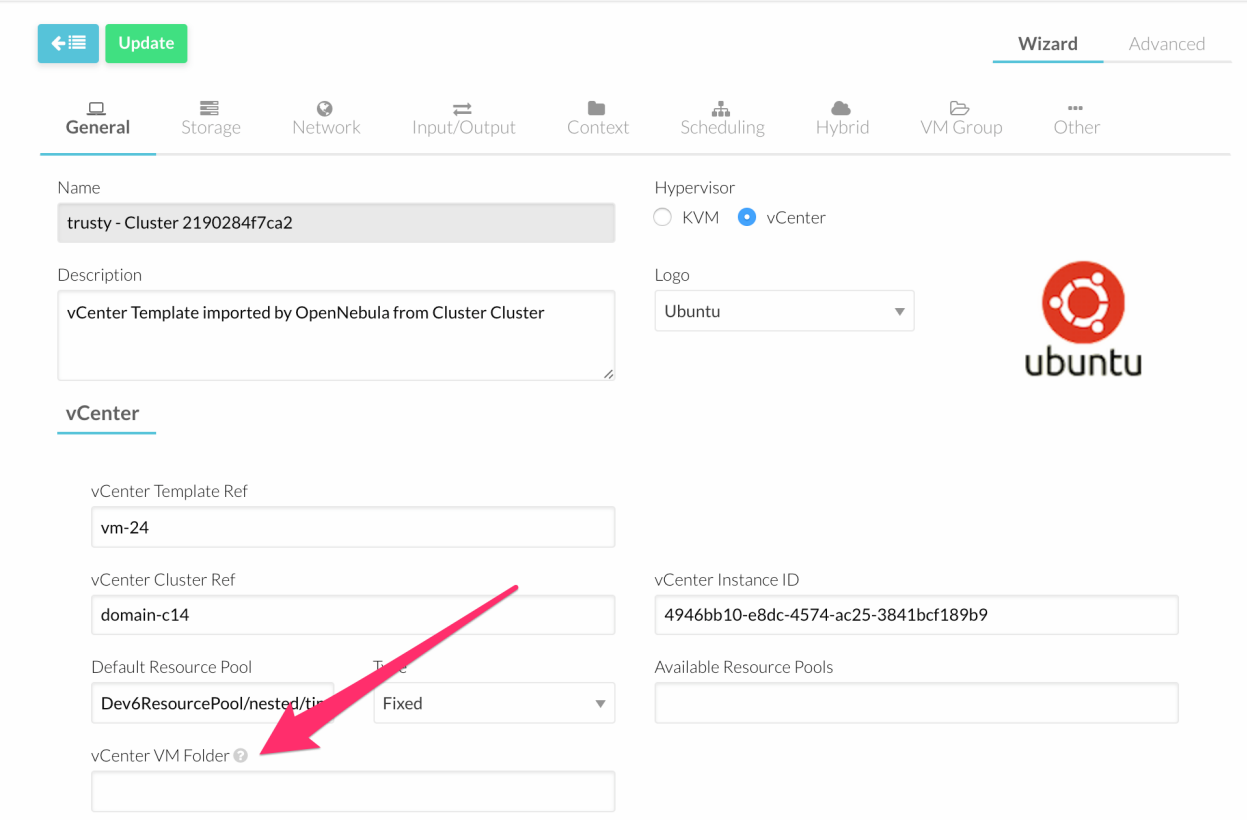
After a VM has been launched, end users in Cloud View can chose to save a VM into a VM Template, provided the VM is in poweroff state. A new VM Template will be created that, upon instantation, will hold all the changes performed to the original VM at the time of hitting the Save As Template button.



VM Placement - Folders

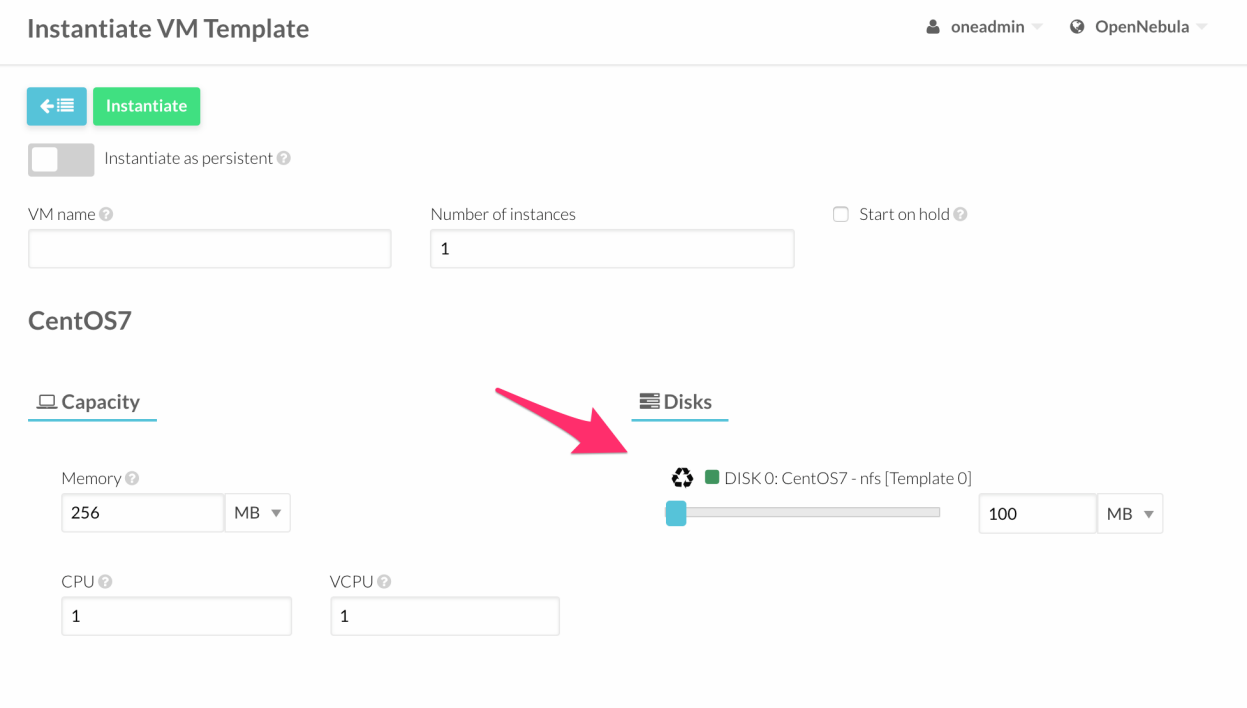
In OpenNebula, by default, a new virtual machine cloned from a vCenter template will be displayed in the same folder where the template lives in vSphere's VM and Templates inventory. However you have the chance to select in which folder you want to see the VM's based on that template.

For example, if you have the following directory tree and you want VMs to be placed in the VMs folder under Management, the path to that folder from the datacenter root would be /Management/VMs. You can use that path in different vOneCloud actions e.g when a template is imported.



Disk Resize

VM disks can be resized at booth time of the VM, or when the VM is powered off.



Disk Save As

VM Disks can be saved as images for later use when the VMs are powered off, the option is present upon clicking a particualr VM and proceeding to the Storage subtab. **Linked Clones**

In vOneCloud, a new VM is deployed when a clone of an existing vCenter template is created, that's why vOneCloud requires that templates are first created in vCenter and then imported into vOneCloud.

In VMWare there are two types of cloning operations:

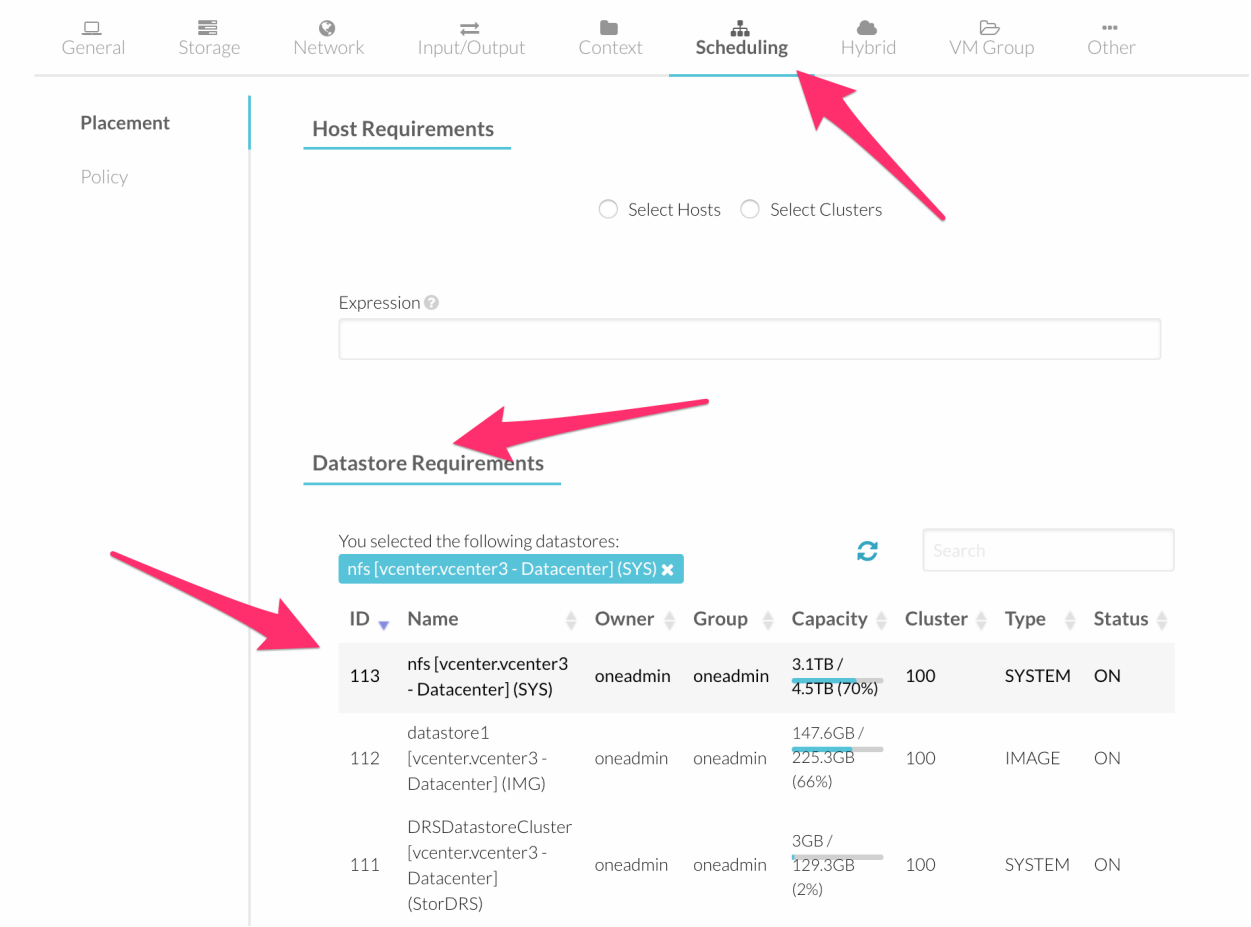
- The Full Clone. A full clone is an independent copy of a template that shares nothing with the parent template after the cloning operation. Ongoing operation of a full clone is entirely separate from the parent template. This is the default clone action in OpenNebula.
- The Linked Clone. A linked clone is a copy of a template that shares virtual disks with the parent template in an ongoing manner. This conserves disk space, and allows multiple virtual machines to use the same software installation.

In order to activate the linked clones functionality VM Templates **must** be imported through the "onevcenter" command line tool, as this functionality is not present through the web interface. To achieve this *login* into the appliance and then use the onevcenter tool as described here, answerying "yes" when prompted if you want to use linked clones. This operation will modify the template so you may prefer that OpenNebula creates a copy of the template and modify that template instead, the onevcenter tool will allow you to choose what you prefer to do.

Warning: Linked clone disks cannot be resized.

Select Datastore

The vOneCloud scheduler will pick a valid datastore for a VM, unless the VM Template defines a datastore. To assign a VM Template to a datastore, proceed to the Scheduling tab on the VM Template update dialog and select the desire datastore in the Datastore requirements section.



Advanced VM Template Editing

The Advanced tab in the VM Template creation / update dialog of vOneCloud, can be used to quickly edit any aspect of the VM Template. The list of attributes that can be used to create / update 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.
SCHED_REQUIREMENTS	Name of the vCenter cluster where this VM Template can instantiated into a VM".
CONTEXT	All sections will be honored except FILES
VCENTER_RESOURCE_POOL	By default, VM will be deployed to the default Resource Pool. This attribute allows to set the name of the resource pool where this VM will be deployed. This can be overwritten explicitly at deployment time from the CLI or Sunstone. More information *here <resource_pool>*.

VM Dynamic Reconfiguration

The following operations can be performed on a running or powered off VM, and the changes will be applied dynamically to the VM:

- Change CPU
- Change Memory
- Add/Remove disks
- Add/Remove NICs
- Change contextualization values

7.2.3 Add New Network/Distributed vSwitch

Virtual Networks from vCenter can be represented using vOneCloud virtual networks, where a one-to-one relationship exists between an vOneCloud's virtual network and a vSphere's port group. When adding NICs in a VM template or when attaching a NIC (hot-plugging) to a running VM in vOneCloud, a network interface can be attached to an vOneCloud's Virtual Network.

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 Import button in the `Network --> Virtual Networks` tab in the vCenter View.

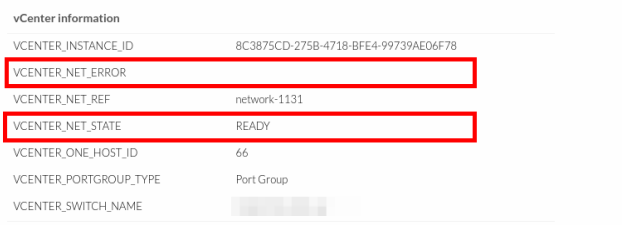
Also, these two kind of networks can be created from vOneCloud.

Creating Port Groups from OpenNebula

This is the workflow when you want vOneCloud to create a vCenter network, regardles of it being a standard or distributed one:

1. Create a new OpenNebula Virtual Network template. Add the required attributes to the template including the OpenNebula's Host ID which represents the vCenter cluster where the network elements will be created.

2. When the Virtual Network is created, vOneCloud will create the network elements required on each ESX host that are members of the specified vCenter cluster.
3. The Virtual Network will be automatically assigned to the vOneCloud cluster which includes the vCenter cluster represented as an vOneCloud host.
4. vOneCloud network creation works asynchronously so you may have to refresh the Virtual Network information until you find the VCENTER_NET_STATE attribute. If it completes the actions successfully that attribute will be set to READY and hence you can use it from VMs and templates. If the network creation task fails VCENTER_NET_STATE will be set to ERROR and the VCENTER_NET_ERROR attribute will offer more information.
5. When a Virtual Network is removed, for each ESX host found in the vCenter cluster assigned to the template, vOneCloud removes both the port group and the switch. If the switch has no more port groups left then the switch will be removed too.



Warning: If a port group or switch is in use e.g a VM is running and have a NIC attached to that port group the remove operation will fail so please ensure that you have no VMs or templates using that port group before trying to remove the Virtual Network representation.

vCenter Network attributes

Here's the table with the attributes that a virtual network representation in vOneCloud understands:

Attribute	Type	Mandatory	Description
VN_MAD	string	Yes	Must be set to vcenter
BRIDGE	string	Yes	It's the port group name.
PHYDEV	string	No	If you want to assign uplinks to your switch you can specify the names of the physical network interface cards of your ESXi hosts that will be used. You can use several physical NIC names using a comma between them e.g vmnic0,vmnic1. Note that two switches cannot share the same physical nics and that you must be sure that the same physical interface name exists and it's available for every ESX host in the cluster. This attribute will be ignored if the switch already exists.
VCENTER_PORTGROUP_TYPE	string	Yes	There are two possible values Port Group and Distributed Port Group. Port Group means a Standard Port Group
VCENTER_ONE_HOST_ID	integer	Yes	The OpenNebula host id which represents the vCenter cluster where the nework will be created.
VCENTER_SWITCH_NAME	string	Yes	The name of the virtual switch where the port group will be created. If the vcenter switch already exists it won't update it to avoid accidental connectivity issues
VCENTER_SWITCH_NPORTS	integer	No	The number of ports assigned to a virtual standard switch or the number of uplink ports assigned to the Uplink port group in a Distributed Virtual Switch. This attribute will be ignored if the switch already exists.
MTU	integer	No	The maximum transmission unit setting for the virtual switch. This attribute will be ignored if the switch already exists.
VLAN_ID	integer	Yes (unless AUTOMATIC_VLAN_ID)	The VLAN ID, will be generated if not defined and AUTOMATIC_VLAN_ID is set to YES
AUTOMATIC_VLAN_ID	boolean	Yes (unless VLAN_ID)	Mandatory and must be set to YES if VLAN_ID hasn't been defined so OpenNebula created a VLAN ID automatically

OpenNebula uses the following values when creating virtual switches and port groups in vCenter according to what the vSphere's Web Client uses in the same operations:

- VLAN ID is set to 0, which means that no VLANs are used.
- MTU value is set to 1500.

Standard port groups created by OpenNebula have the following settings:

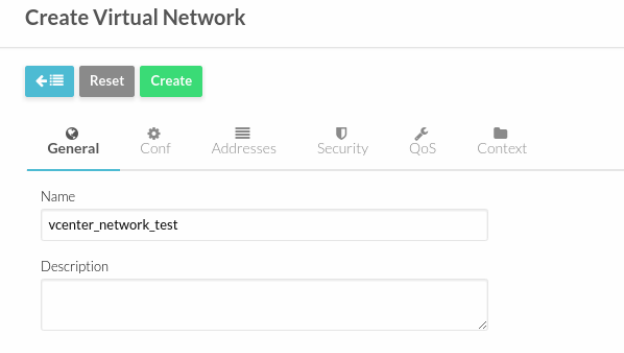
- Number of ports is set to Elastic. According to VMWare's documentation, the Elastic mode is used to ensure efficient use of resources on ESXi hosts where the ports of virtual switches are dynamically scaled up and down. In any case, the default port number for standard switches is 128.
- Security - Promiscuous mode is set to Reject, which means that the virtual network adapter only receives frames that are meant for it.
- Security - MAC Address Changes is set to Accept, so the ESXi host accepts requests to change the effective MAC address to other than the initial MAC address.
- Security - Forged transmits is set to Accept, which means that the ESXi host does not compare source and effective MAC addresses.
- Traffic Shaping policies to control the bandwidth and burst size on a port group are disabled. You can still set QoS for each NIC in the template.

- Physical NICs. The physical NICs used as uplinks are bridged in a bond bridge with teaming capabilities.

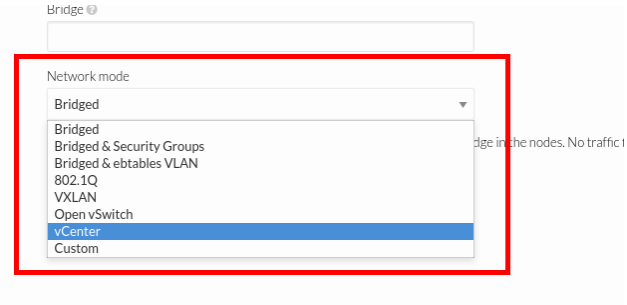
Distributed port groups created by OpenNebula have the following settings:

- Number of ports is set to Elastic. According to VMWare's documentation, the Elastic mode is used to ensure efficient use of resources on ESXi hosts where the ports of virtual switches are dynamically scaled up and down. The default port number for distributed switches is 8.
- Static binding. When you connect a virtual machine to a distributed port group, a port is immediately assigned and reserved for it, guaranteeing connectivity at all times. The port is disconnected only when the virtual machine is removed from the port group.
- Auto expand is enabled. When the port group is about to run out of ports, the port group is expanded automatically by a small predefined margin.
- Early Bindind is enabled. A free DistributedVirtualPort will be selected to assign to a Virtual Machine when the Virtual Machine is reconfigured to connect to the port group.

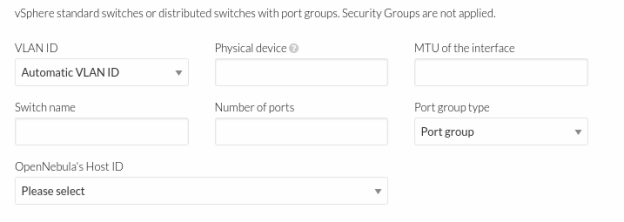
A sample session to create a Virtual Network follow. The first step requires you to introduce the virtual network's name:



In the Conf tab, select vCenter from the Network Mode menu, so the vcenter network driver is used (the `VN_MAD=vcenter` attribute will be added to OpenNebula's template). The Bridge name will be the name of the port group, and by default it's the name of the Virtual Network but you can choose a different port group name.



Once you've selected the vCenter network mode, Sunstone will show several network attributes that can be defined.



Address Ranges

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*.

Traffic Shaping

In order to get VM traffic shaping to work, the NIC must be controlled by vOneCloud and it needs to be connected to a Distributed vSwitch. The following requirements also needs to be met:

- Verify that vSphere Distributed Switch is version 6.0.0 and later.
- Verify that Network I/O Control on the switch is version 3.
- Verify that Network I/O Control is enabled.
- Verify that the virtual machine system traffic has a configured bandwidth reservation.

Steps to achieve the above configuration can be found here

Four values can be used in both the Virtual Network Template or the NIC to achieve traffic shaping. Take into account that only total traffic (inbound and outbound) is limited, the minimum between inbound and outbound is picked.

- Minimum between **INBOUND_AVG_BW** and **OUTBOUND_AVG_BW**. Expressed in kilobytes/second, this value is used to set the Reservation. This value cannot be set to a greater value than the Peak_BW.
- Minimum between **INBOUND_PEAK_BW** and **OUTBOUND_PEAK_BW**. Expressed in kilobytes/second, this value is used to set the limit, or maximum bitrate for the interface of the VM. This value cannot be less than 1024 kilobytes/second.

Network Monitoring

vOneCloud gathers network monitoring info for each VM. Real-time data is retrieved from vCenter thanks to the Performance Manager which collects data every 20 seconds and maintains it for one hour. Real-time samples are used so no changes have to be applied to vCenter's Statistics setings. Network metrics for transmitted and received traffic are provided as an average using KB/s unit.

The graphs provided by Sunstone are different from those found in vCenter under the Monitor -> Performance Tab when selecting Realtime in the Time Range drop-down menu or in the Advanced view selecting the Network View. The reason is that Sunstone uses polling time as time reference while vCenter uses sample time on their graphs, so an approximation to the real values aggregating vCenter's

7.2.4 Add New Datastore

Datastores 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 Import button in the `Storage --> Datastores` tab.

In order to create a OpenNebula vCenter datastore that represents a vCenter VMFS datastore, a new OpenNebula datastore needs to be created with the following attributes. This can be achieved using the '+' sign in `Storage --> Datastores` tab.

Attribute	Description
DS_MAD	Must be set to vcenter if TYPE is SYSTEM_DS
TM_MAD	Must be set vcenter
TYPE	Must be set to SYSTEM_DS or IMAGE_DS
VCENTER_ADAPTER_TYPE	Default adapter type used by virtual disks to plug inherited to VMs for the images in the datastore. It is inherited by images and can be overwritten if specified explicitly in the image. Possible values (careful with the case): lsiLogic, ide, busLogic. More information in the VMware documentation. Known as "Bus adapter controller" in Sunstone.
VCENTER_DISK_TYPE	Type of disk to be created when a DATABLOCK is requested. This value is inherited from the datastore to the image but can be explicitly overwritten. The type of disk has implications on performance and occupied space. Values (careful with the case): delta,eagerZeroedThick,flatMonolithic,preallocated,raw,rdm,rdmp,seSparse,sparse2Gb,sparseMonolithic,thick,thick2Gb,thi More information in the VMware documentation. Known as "Disk Provisioning Type" in Sunstone.
VCENTER_DS_REF	Managed Object Reference of the vCenter datastore. Please visit the Managed Object Reference section to know more about these references.
VCENTER_DC_REF	Managed Object Reference of the vCenter datacenter. Please visit the Managed Object Reference section to know more about these references.
VCENTER_INSTANCE_ID	The vCenter instance ID. Please visit the Managed Object Reference section to know more about these references.
VCENTER_HOST	Hostname or IP of the vCenter host
VCENTER_USER	Name of the vCenter user.
VCENTER_PASSWORD	Password of the vCenter user. It's encrypted when the datastore template is updated using the secret stored in the /var/lib/one/.one/one_key an
VCENTER_CLUSTER	(Optional) Specifies what folder under the root directory of the datastore will host persistent and non-persistent images e.g one
VCENTER_IMPORTED	(Optional) Specifies what folder under the root directory of the datastore will host the volatile disks

All OpenNebula datastores are actively monitoring, and the scheduler will refuse to deploy a VM onto a vCenter datastore with insufficient free space.

7.2.5 Add New Images / CDROMS

Adding a new datastore and representing existing VMDK images enables disk attach/detach functionality.

There are three ways of adding VMDK representations in vOneCloud:

- Upload a new VMDK from the local filesystem
- Register an existent VMDK image already in the datastore
- Create a new empty datablock

vCenter VM Templates with already defined disks will be imported without this information in vOneCloud. These disks will be invisible for vOneCloud, and therefore cannot be detached from the VMs. The imported Templates in vOneCloud can be updated to add new disks from VMDK images imported from vCenter (please note that these will always be persistent).

The following image template attributes need to be considered for vCenter VMDK image representation in vOneCloud:

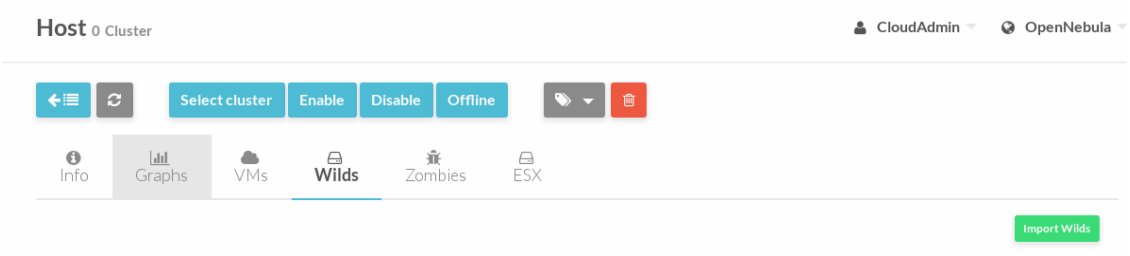
Attribute	Description
PERSISTENT	At the time of instantiating a VM with a disk using this image as backing, wheter vOneCloud will copy this image (Set to NO) or if the original will be used (Set to YES)
PATH	This can be either: • local filesystem path to a VMDK to be uploaded, which can be a single VMDK or a tar of vmdk descriptor and flat files which can be uncompressed or compressed with gzip or bzip2 (no OVAs supported) • path of an existing VMDK file in the vCenter datastore. In this case a "vcenter://" prefix must be used (for instance, an image win10.vmdk in a Windows folder should be set to vcenter://Windows/win10.vmdk)
ADAPTER_TYPE	Possible values (careful with the case): lsiLogic, ide, busLogic. More information in the VMware documentation. Known as "Bus adapter controller" in Sunstone.
DISK_TYPE	The type of disk has implications on performance and occupied space. Values (careful with the case): delta,eagerZeroedThick,flatMonolithic,preallocated,raw,rdm,rdmp,seSparse More information in the VMware documentation

VMDK images in vCenter datastores can be:

- Cloned
- Deleted
- Hotplugged to VMs

7.3 Import Running and Powered Off 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.



VNC capacibilities will be automatically add to imoported VMs.

In the ZOMBIES tab you'll find VMs that were launched from OpenNebula but, for whatever reason, OpenNebula is not aware of this, e.g coming from a different OpenNebula installation, or being managed from a different vOneCloud. Zombie VMs are meant to be a warning of a VM that need manual clean-up.

Read more about the vCenter drivers. Regarding the vCenter datastores in vOneCloud, refer to the vCenter datastore guide

7.4 Storage DRS and datastore cluster

Thanks to vOneCloud's scheduler, you can manage your datastores clusters with load distribution but you may already be using vCenter's Storage DRS capabilities. Storage DRS allows you to manage the aggregated resources of a datastore cluster. If you're using Storage DRS, vOneCloud can delegate the decision of selecting a datastore to the Storage DRS cluster (SDRS) but as this behavior interferes with vOneCloud's scheduler and vSphere's API impose some restrictions, there will be some limitations in StorageDRS support in vOneCloud.

When you import a SDRS cluster using onevcenter or Sunstone:

- The cluster will be imported as a SYSTEM datastore only.
- vOneCloud detects the datastores grouped by the SDRS cluster so you can still import those datastores as both IMAGE and SYSTEM datastores.
- Non-persistent images are not supported by a SDRS as vSphere's API does not provide a way to create, copy or delete files to a SDRS cluster as a whole, however you can use persistent and volatile images with the VMs backed by your SDRS.
- Linked clones over SDRS are not supported by vOneCloud, so when a VM clone is created a full clone is performed.

In order to delegate the datastore selection to the SDRS cluster you must inform vOneCloud's scheduler that you want to use specifically the SYSTEM datastore representing the storage cluster. You can edit a VM template and select the storage cluster in the Scheduling tab.

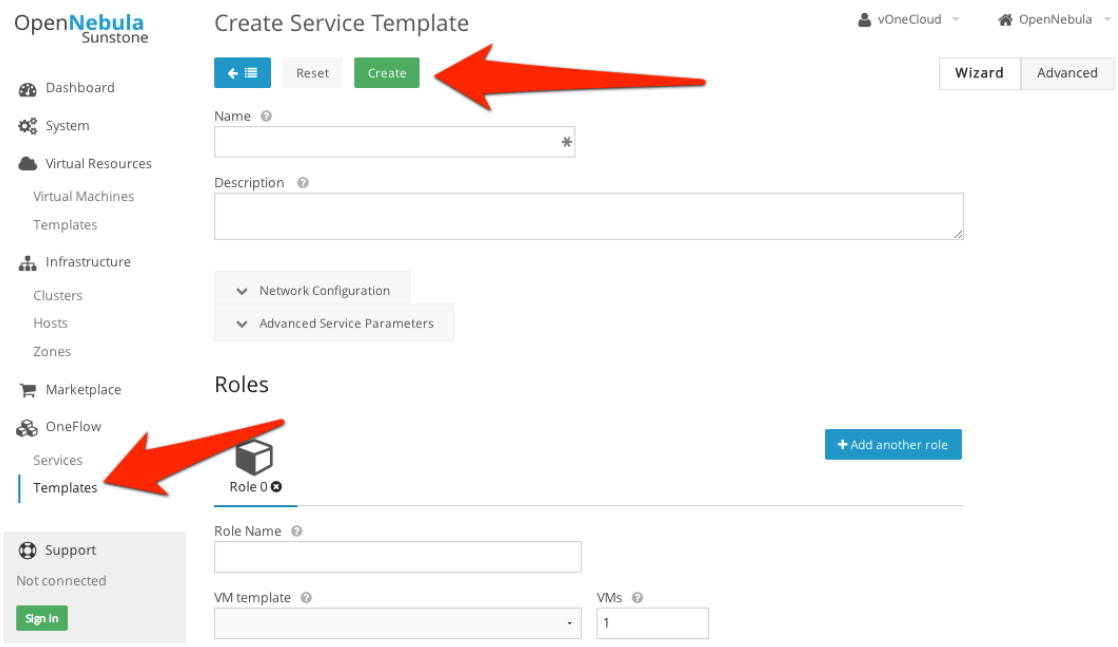
Current support has the following limitations:

- Images in StoragePods can't be imported through Sunstone although it's possible to import them from a datastore, which is a member of a storage cluster, if it has been imported previously as an individual datastore.
- New images like VMDK files cannot be created or uploaded to the StoragePod as it's set as a SYSTEM datastore. However, it's possible to create an image and upload it to a datastore which is a member of a storage cluster it has been imported previously as an individual datastore.

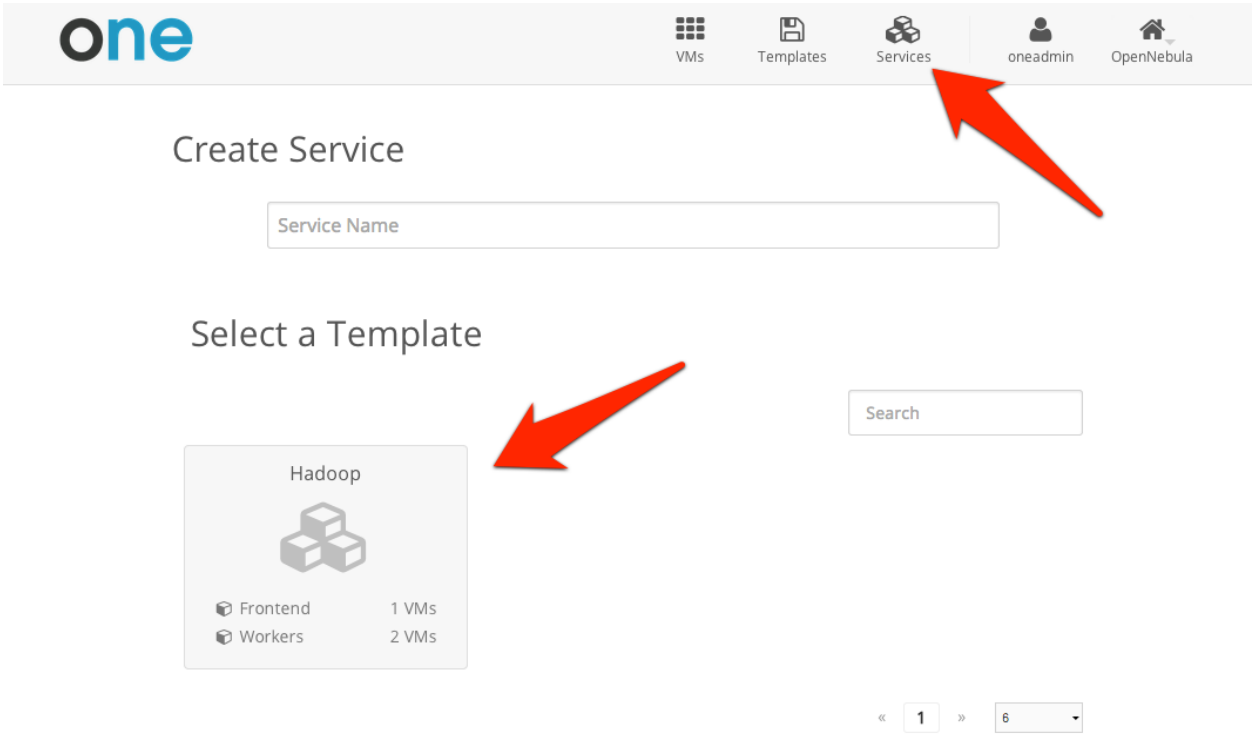
7.5 Multi VM Applications

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.

7.6 Virtual Routers

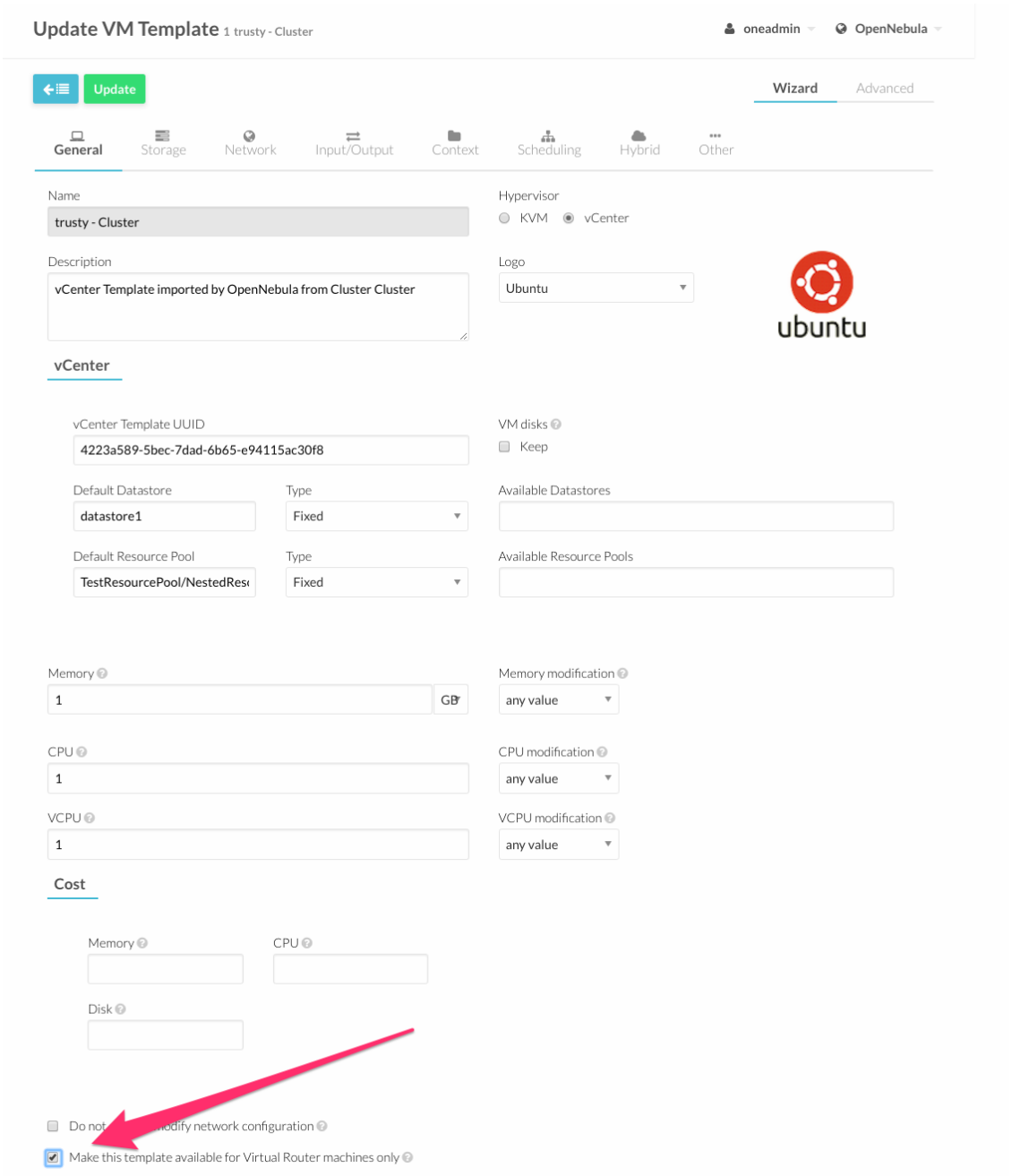
Virtual Routers provide routing across Virtual Networks. Routing is implemented with a Virtual Machine appliance.

7.6.1 Creating Virtual Routers

An OVA containing the Virtual Router is available as an OVA which can be imported from the following URL:

https://s3-eu-west-1.amazonaws.com/opennebula-marketplace/alpine-quagga.ova

The OVA needs to be registered as a VM Template directly in vCenter. Afterwards the VM Template can be imported in vOneCloud, and set the template as a Virtual Router at the bottom of the General tab of the vOneCloud VM Template update wizard.



Virtual Routers can be seamlessly deployed in high availability mode.

To create a new Virtual Router, proceed to `Instances --> Virtual Routers` and click on the **+** icon. Follow the wizard to select the Virtual Networks that will get logically linked to it. This connection takes effect when the Virtual Machine containing the VR Appliance is automatically deployed, with a network interface attached to each Virtual Network.

For each Virtual Network, the following options can be defined:

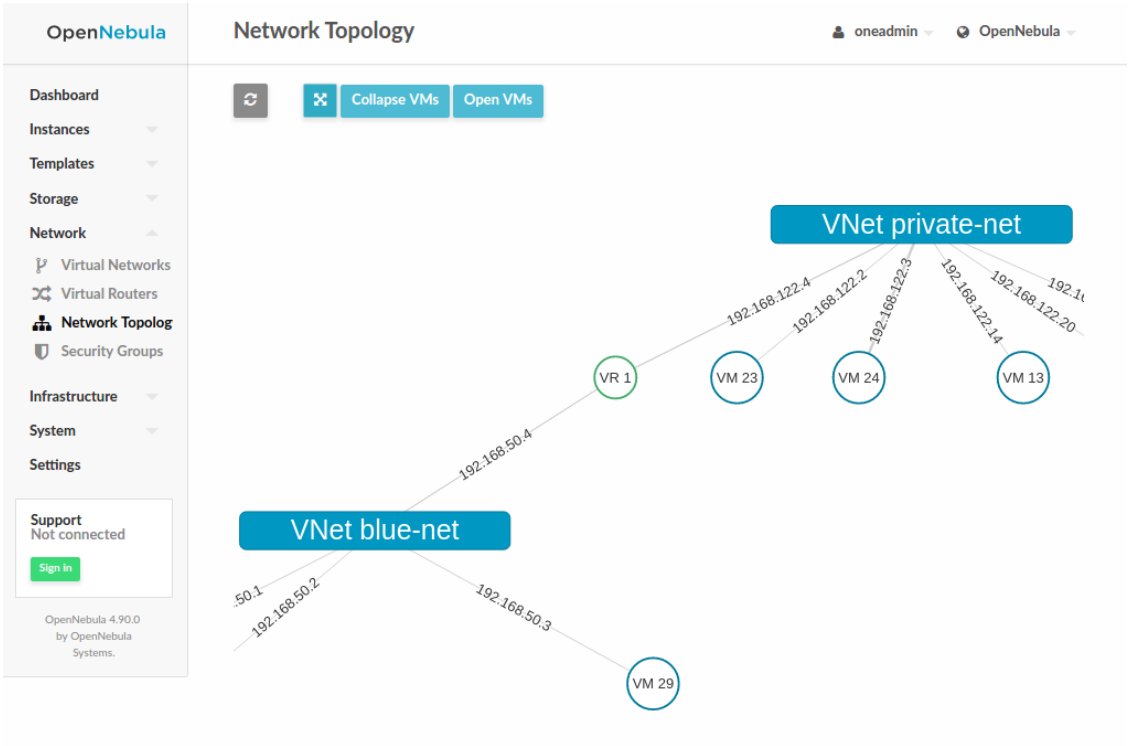
- **Floating IP**. Only used in High Availability, explained below.
- **Force IPv4**. You can force the IP assigned to the network interface. When the VR is not configured in High Availability, this will be the IP requested for the Virtual Machine appliance.

- **Management interface**. If checked, this network interface will be a Virtual Router management interface. Traffic will not be forwarded to it.

Once ready, click the "create" button to finish. vOneCloud will create the Virtual Router and the Virtual Machines automatically.

7.6.2 Managing Virtual Routers

Using the Virtual Routers tab you can retrieve the generic resource information such as owner and group, the list of Virtual Networks interconnected by this router, and the Virtual Machines that are actually providing the routing.



The Virtual Networks connected to the VR machines can be modified with the attach/detach actions.

Actions can be found in the Virtual Router's main information panel, in the networks table. The options to add a new Virtual Network are the same that were explained for the creation wizard, see previous section.

After a NIC is attached or detached, the Virtual Machine appliances are automatically reconfigured to start routing to the new interface. No other action, like a reboot, is required.

Virtual Machines associated to a Virtual Router have a limited set of actions. Specifically, any action that changes the VM state cannot be executed, including terminate.

To terminate a Virtual Machine associated with a Virtual Router, you need to delete the Virtual Router.

7.6.3 High Availability

More than one Virtual Machines can be associated to a Virtual Router in order to implement a high availability scenario. In this case, vOneCloud will also assign a floating IP to the group of Virtual Machines, that will coordinate to manage the traffic directed to that IP.

To enable a high availability scenario, you need to choose 2 or more number of instances when the Virtual Router is created. In this scenario, the following Virtual Router options became relevant:

- **Keepalived ID**: Optional. Sets keepalived configuration parameter `virtual_router_id`.
- **Keepalived password**: Optional. Sets keepalived configuration parameter `authentication/auth_pass`.

And for each Virtual Network Interface:

- **Floating IP**. Check it to enable the floating IP.
- **Force IPv4**. Optional. With the floating IP option selected, this field requests a fixed IP for that floating IP, not the individual VM IPs.

The floating IP assignment is managed in a similar way to normal VM IPs. If you open the information of the Virtual Network, it will contain a lease assigned to the Virtual Router (not a VM). Besides the floating IP, each VM will get their own individual IP.

Other Virtual Machines in the network will use the floating IP to contact the Virtual Router VMs. At any given time, only one VM is using that floating IP address. If the active VM crashes, the other VMs will coordinate to assign the floating IP to a new Virtual Router VM.

7.6.4 Customization

You can provide two optional parameters in the context to configure the keepalived service started in the Virtual Router VM:

- `VROUTER_KEEPALIVED_PASSWORD`: Password used for the service to protect the service from packages of rogue machines. By default the service is configured without password.
- `VROUTER_KEEPALIVED_ID`: Number identifier of the service (0-255). This is useful when you have several virtual routers or other keepalived services in the same network. By default it is generated from the Virtual Router ID (`$vrouter_id & 255`) but you can specify it manually if needed.

These parameters can also be provided in the Virtual Router creation wizard.

7.7 Authentication

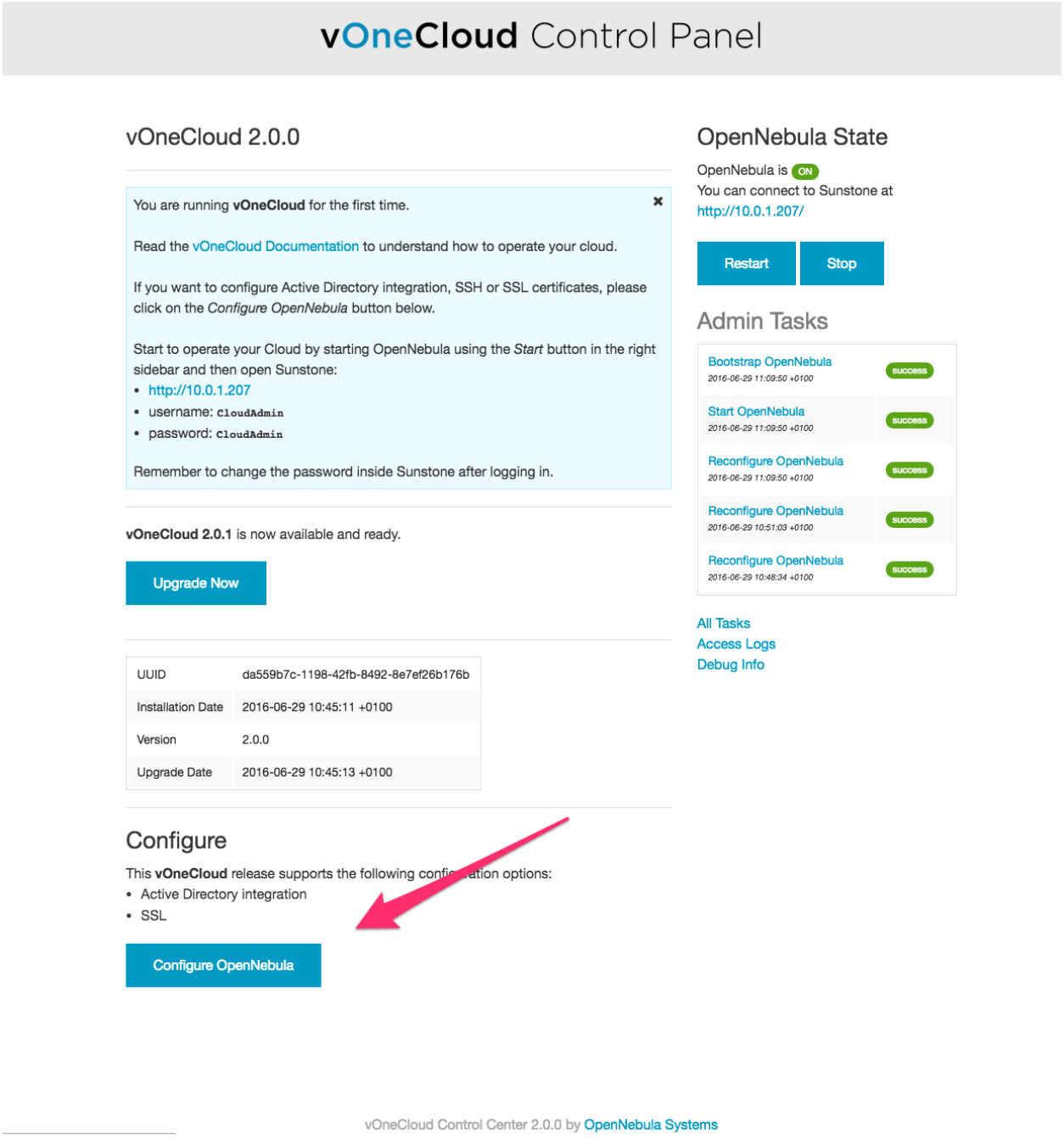
By default, vOneCloud authentication uses an internal user/password system with user and group information stored in an internal database.

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.

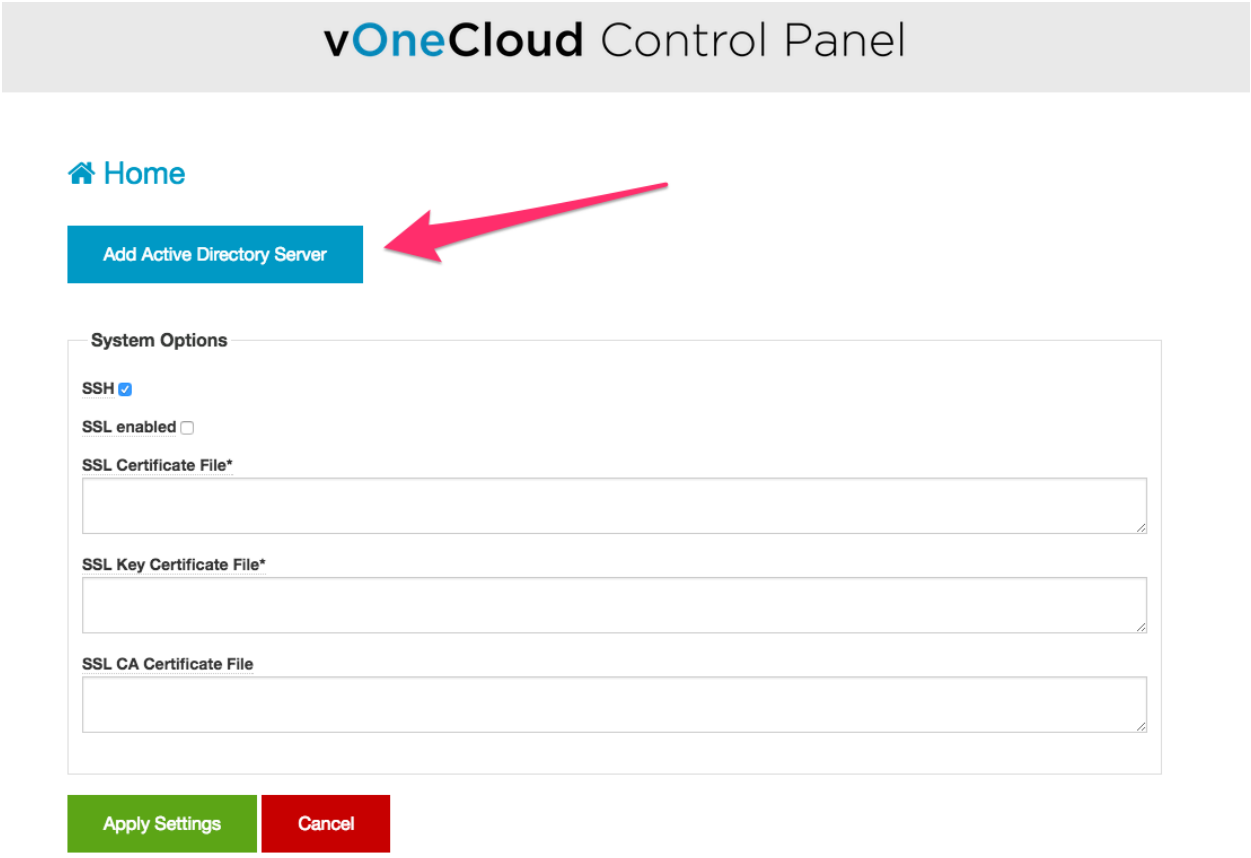
You will need to access the *Control Panel* in order to configure the Active Directory support in vOneCloud. After the configuration is done, users that exist in Active Directory can begin using vOneCloud.

7.7.1 Step 1. Configure Active Directory support

Click on the "Configure OpenNebula" button



In the following screen, select the “Add Active Directory” category



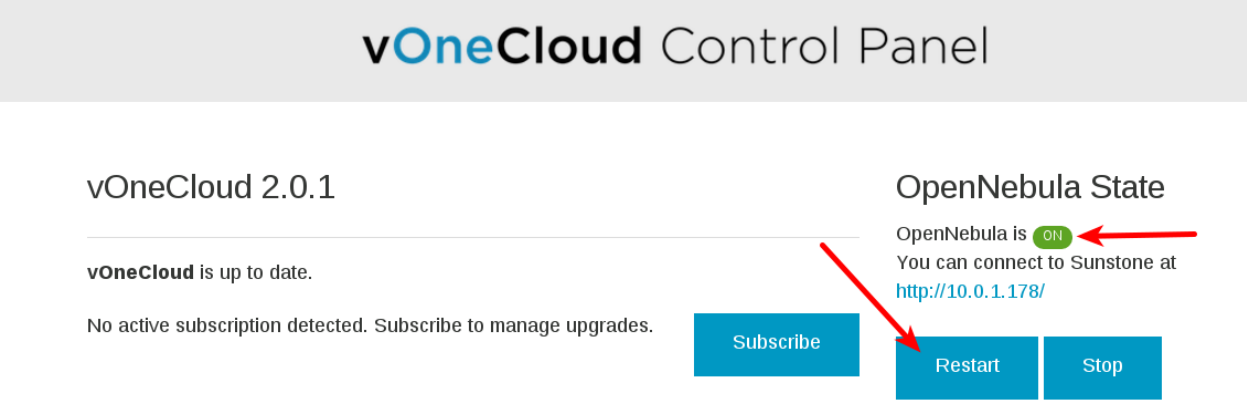
Fill the needed fields following the criteria described in the next table

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.

7.7.2 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 information on the integration with Active Directory in this guide.

7.8 Resource Pool Confinement

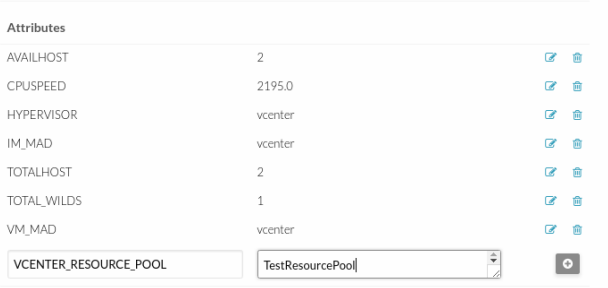
vOneCloud can place VMs in different Resource Pools. There are two approaches to achieve this:

- fixed per Cluster basis
- flexible per VM Template basis.

7.8.1 Fixed per Cluster basis

In the fixed per Cluster basis approach, the vCenter connection that vOneCloud use 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.
- Import the vCenter cluster into vOneCloud as explained later. The import action will create an vOneCloud host.
- Add a new attribute called VCENTER_RESOURCE_POOL to vOneCloud's host template representing the vCenter cluster (for instance, in the info tab of the host, or in the CLI), with the reference to a Resource Pool.



7.8.2 Flexible per VM Template

The second approach is more flexible in the sense that all Resource Pools defined in vCenter can be used, and the mechanism to select which one the VM is going to reside into can be defined using the attribute VCENTER_RESOURCE_POOL in the vOneCloud VM Template.

Once we have in vOneCloud an imported template, we can **update it** from the CLI or the Sunstone interface and we will have two choices:

- Specify a fixed Resource Pool that will be used by any VM based on the template.
- Offer a list of Resource Pools so the user can select one of them when a VM is instantiated.

Using the CLI we would use the **onetemplate update** command and we would add or edit the VCENTER_RESOURCE_POOL attribute.

If we want to specify a Resource Pool, that attribute would be placed inside the template and would contain a reference to the resource pool.

```
VCENTER_RESOURCE_POOL="TestResourcePool/NestedResourcePool"
```

If we wanted to offer a list to the user, we would place the VCENTER_RESOURCE_POOL attribute inside a USER_INPUT element, an it would contain a string that represents a list. Let's see an example:

```
USER_INPUTS=[
    VCENTER_RESOURCE_POOL="O|list|Which resource pool you want this VM to run in?␣
↪|TestResourcePool/NestedResourcePool,TestResourcePool|TestResourcePool/
↪NestedResourcePool" ]
```

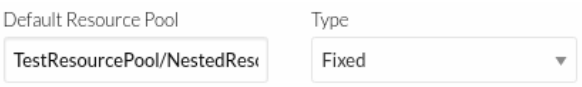
The VCENTER_RESOURCE_POOL has the following elements:

- O: it means that it is optional to select a Resource Pool.
- list: this will be a list shown to users.
- Which resource pool you want this VM to run in?: that's the question that will be shown to users.
- TestResourcePool/NestedResourcePool,TestResourcePool: that's the list of Resource Pool references separeted with commas that are available to the user.
- TestResourcePool/NestedResourcePool: is the default Resource Pool that will be selected on the list.

Note: As we'll see later, the import tools provided by OpenNebula will create the VCENTER_RESOURCE_POOL attribute easily.

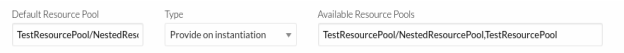
Using Sunstone we have the same actions described for the onevcenter tool.

If we want to specify a Resource Pool we should select Fixed from the Type drop-down menu and introduce the reference under Default Resource Pool:



If we wanted to offer a list to the user:

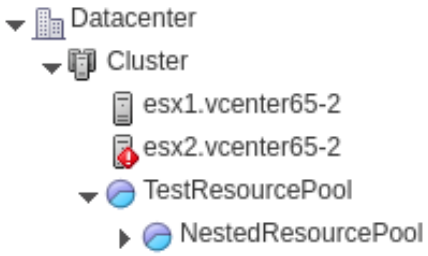
- We would select Provide on Instantiation from the Type drop-down menu.
- We would specify the default value that we want to be selected in the list.
- We would introduce the references of the Resource Pools that we want to include in the list, using a comma to separate values.



7.8.3 Referencing a Resource Pool

The VCENTER_RESOURCE_POOL attribute expects a string containing the name of the Resource Pool. If the Resource Pool is nested, the name of the Resource Pool should be preceeded by slashes and the names of the parent Resource Pools.

For instance, a Resource Pool "NestedResourcePool" nested under "TestResourcePool"



would be represented as "TestResourcePool/NestedResourcePool":

```
VCENTER_RESOURCE_POOL="TestResourcePool/NestedResourcePool"
```

7.9 Resource Deletion

There are different behavior of the vCenter resources when deleted in vOneCloud.

The following resources are NOT deleted in vCenter when deleted in vOneCloud:

- VM Templates.
- Networks. Unless OpenNebula has created the port groups and/or switches instead of just consume them.
- Datastores.

The following resource are deleted in vCenter when deleted in vOneCloud:

- Virtual Machines.
- Images. A VMDK or ISO file will be deleted in vCenter unless the VCENTER_IMPORTED attribute is set to YES.

CHAPTER

EIGHT

APPLIANCE CONFIGURATION

8.1 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.

8.2 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.

8.2.1 Initial Bootstrap

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.

8.2.2 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*.

8.3 Control Panel

This is a web based interface available at `http://<appliance_ip>:8000` which handles many aspects of the vOneCloud platform configuration.

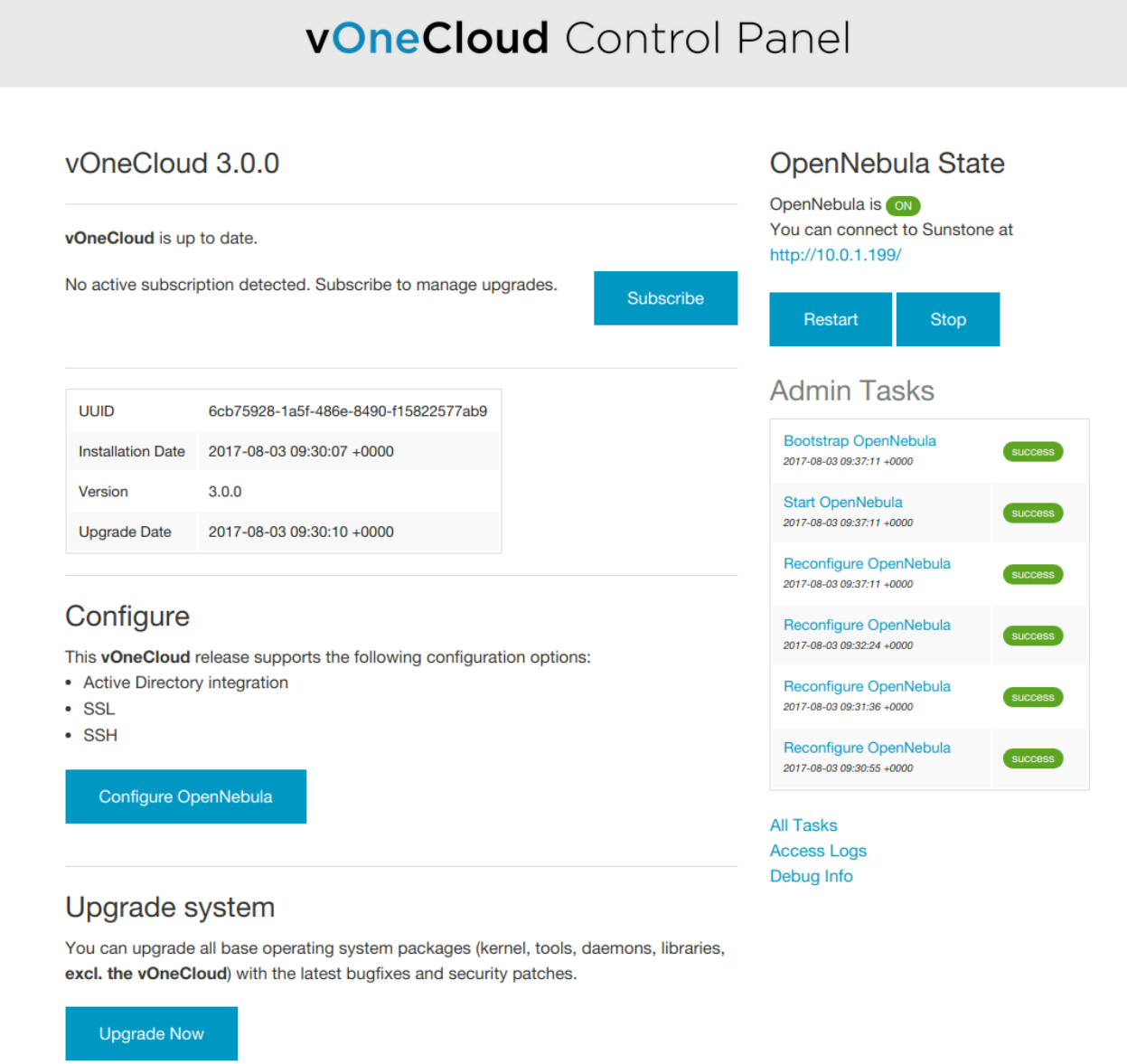
To log in the administrator will need the `oneadmin` account, which is set in the initial configuration of the Control Console.

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

8.3.1 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

8.3.2 Configuration Management

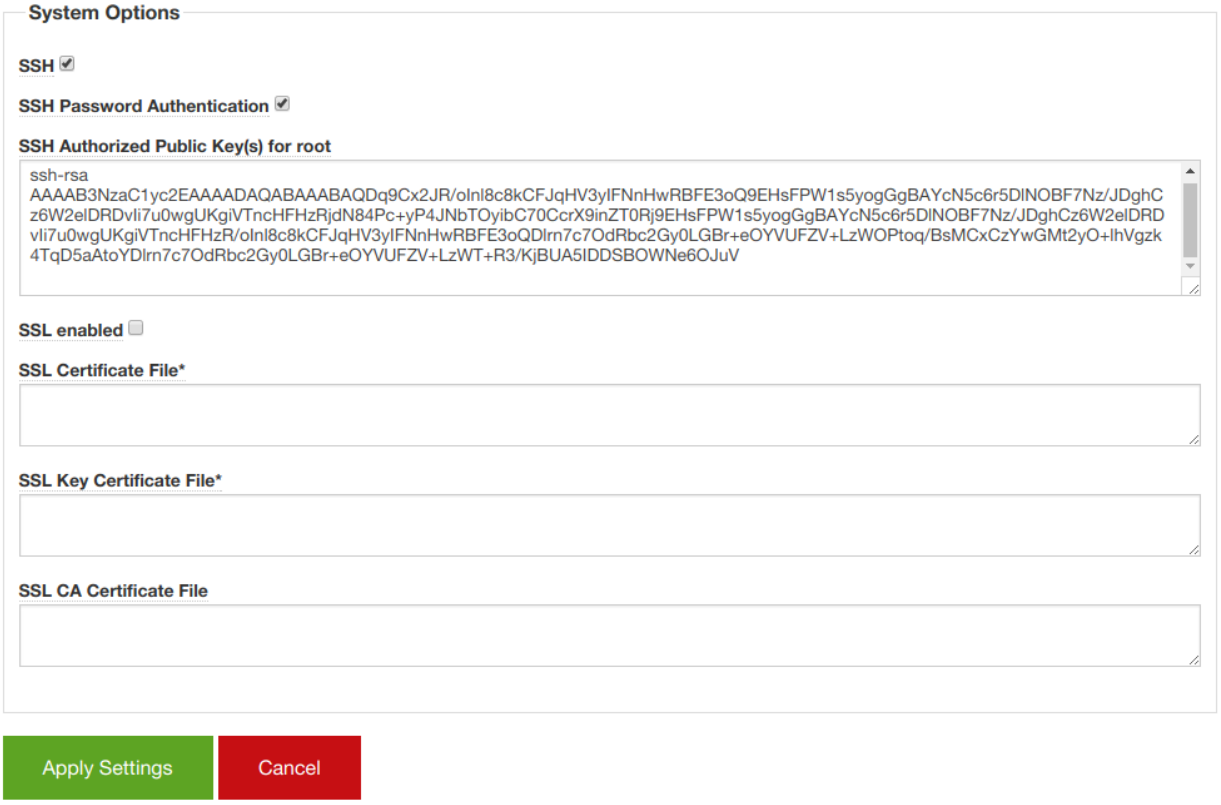
The configuration action handles the supported configuration of the vOneCloud appliance:

- *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.

8.3.3 System options

It is possible to configure SSH and SSL:



SSH

By default SSH access is disabled. If you want to enable it, enable the `SSH Password Authentication` checkbox.

You can choose whether to allow password based authentication. If you only want public ssh key authentication you need to fill in the `SSH Authorized Public Key(s) for root` field.

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)

Note: If you are going to use a self-signed SSL certificate, and do not have the CA certificate, you will need to have your browser trust that certificate, in both 443 and 29876 ports in the vOneCloud IP or FQDN. Otherwise VNC may not work

8.3.4 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.

8.3.5 Log Access

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

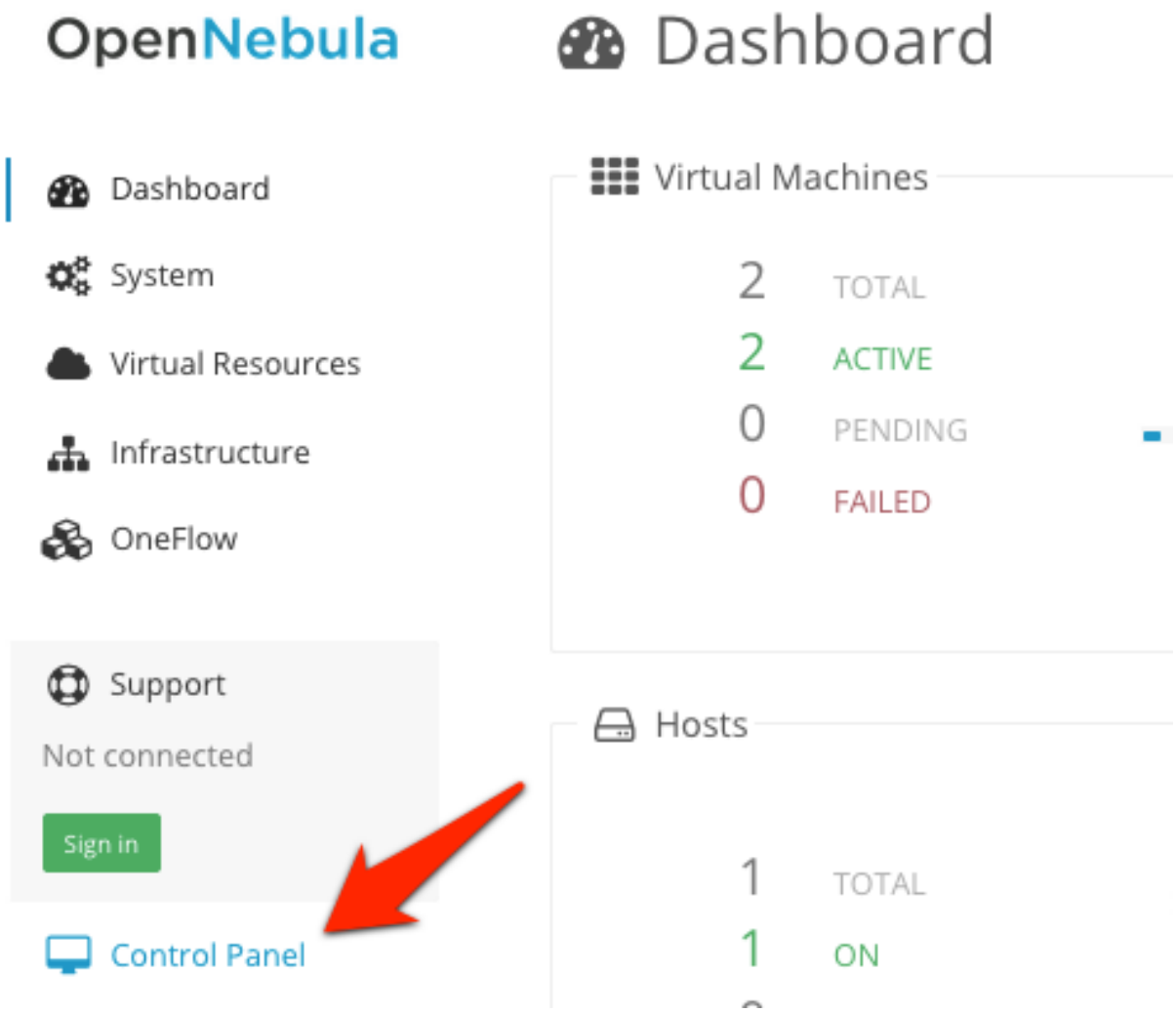
8.3.6 Automatic Upgrades

When a new vOneCloud release is available for download users will be notified both in Sunstone and in the Control Panel. Users 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.

8.3.7 Opening the Control Panel from Sunstone

Once OpenNebula and Sunstone have been started in the Control Panel, you will be able to see a link the Sunstone GUI to go back to the Control Panel. Of course, you can also manually open `http://<appliance_ip>:8000`.



8.4 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.

8.4.1 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.png`. 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="X-UA-Compatible" content="IE=edge" />
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

    <title>OpenNebula Sunstone Login</title>

    <!--[if IE]><link rel="shortcut icon" href="images/favicon.ico"><![endif]-->
    <link rel="apple-touch-icon-precomposed" href="images/apple-touch-icon-
↪precomposed.png">
    <link rel="icon" href="images/favicon.png">

    <link rel="stylesheet" type="text/css" href="css/login.css" />

    <% 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") || (settings.config[:auth] == "remote")
↪%>
    <%= erb :_login_x509 %>
    <% else %>
    <%= erb :_login_standard %>
    <% end %>

    <div id="footer" style="overflow:visible;">
      <a href="http://opennebula.org" target="_blank">OpenNebula 5.4.0</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;
-webkit-background-size: cover;
-moz-background-size: cover;
-o-background-size: cover;
background-size: cover;
```

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

8.4.2 VM Template Logos

It is possible to add new logos for the VM Templates to be displayed in Sunstone:

- Create your logo in PNG format (90 x 96 pixels).
- Log in into the appliance and place it in `/usr/lib/one/sunstone/public/images/logos`.
- Run `chmod +644` on the uploaded file.
- In Sunstone vCenter Admin view, update the desired VM Template and select any of the built in logos.
- Click on update again and switch to Advanced view.
- Change the the `LOGO=` line to `LOGO="images/logos/<mylogo>.png`.

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

8.5 Troubleshooting

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

8.5.1 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:

```
export HTTPS_PROXY=http://<proxy_user>:<proxy_pass>@<proxy_host>:<proxy_port>
curl -kv  https://downloads.vonecloud.com/version
```

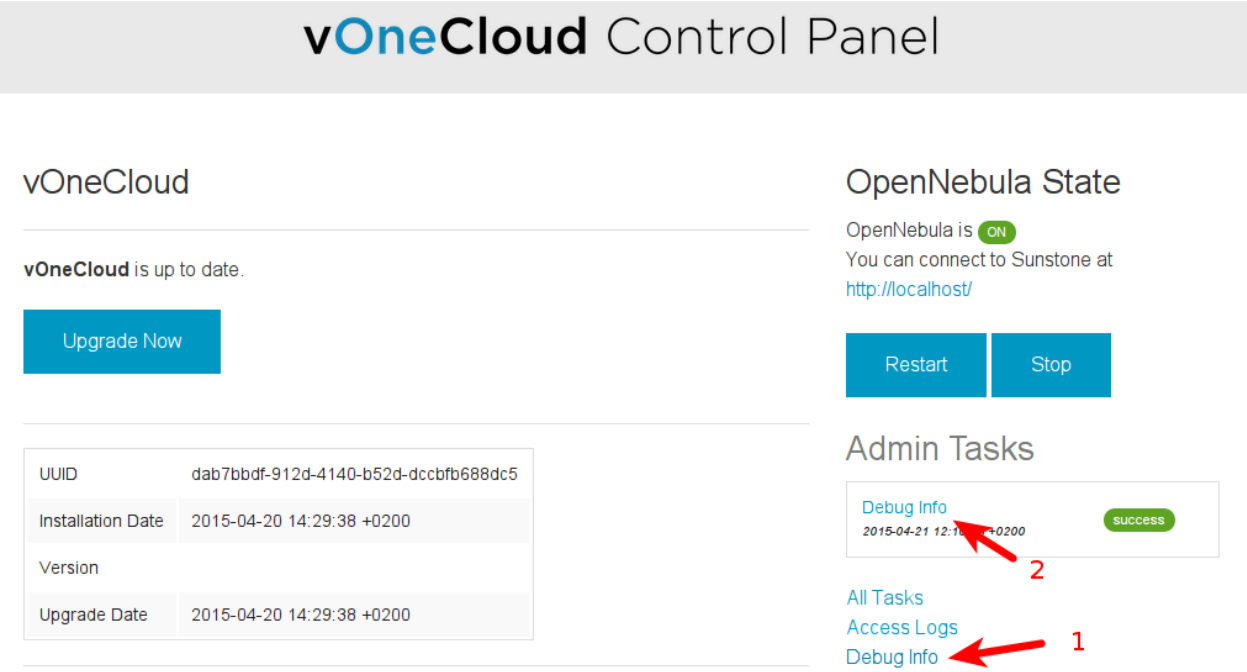
If you are sure the network is properly configured, please feel free to submit a support to vOneCloud Support.

8.5.2 Debug Information

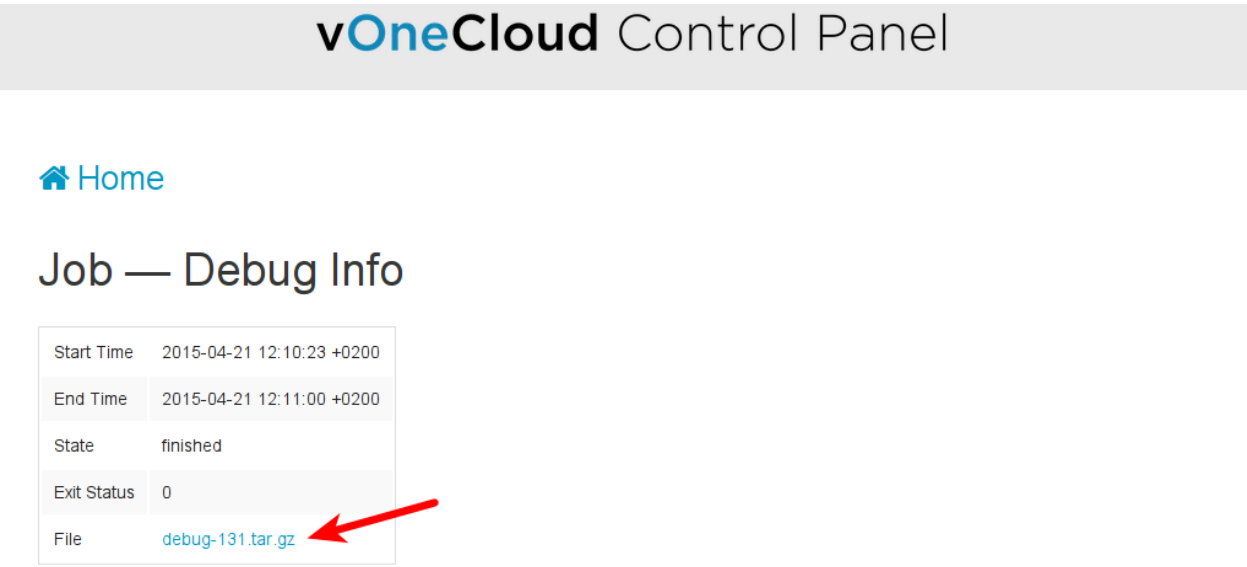
An Admin Task called `Debug Info` generates a gzipped tar file which can be downloaded that contains all the required information to debug the cloud if the OpenNebula user runs into a problem. This file can be then sent to vOneCloud Support. Note that this sends information on all the resources of the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



8.5.3 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.