UG3.5: ADA Cloud UserGuide

In this page:

- General information
- System Architecture
- Cloud Model
- Service model
- Flexible authentication model
- Roles and responsibilities
- Quotas, Flavors and Images
- Access
- Creation of an instance of the virtual machine in your Project
- Operations with an instance
    - Create an instance snapshot
    - Manage an instance
    - Track usage for instances
    - Monitor instances
    - Rescue a Virtual Machine
    - Resize a VM (a.k.a. change its flavor)
    - Resize of a VM's Bootable Volume
    - Download a VM
- Cinder Volumes
- Sharing a filesystem among virtual machines: the Manila service
- Storing of sensitive data

Additional page:

- FAQ for ISCRA project submission
- Command Line Interface (CLI) - OpenStack Application Credential
- Manila OpenStack service - How to share filesystem in OpenStack
- Trove Openstack Service
- Virtual Machine administration tips
- General Security Guidelines for ADA cloud

General information

early availability: 05 Aug 2021

start of pre-production: 01 Sep 2021

start of production: 27 Sep 2021

Model: DUal-Socket Dell PowerEdge

Cloud Platform: OpenStack version Zed
Nodes: 71
Processors:2xCPU 8260 Intel CascadeLake (24c, 2.4Ghz)
Cores: 48 cores/node, Hyperthreading x2
RAM: 768GB DDR4
Internal Network: Ethernet 100GbE



System Architecture

The HPC cloud infrastructure, named ADA cloud is based on OpenStack Zed.

Provides:

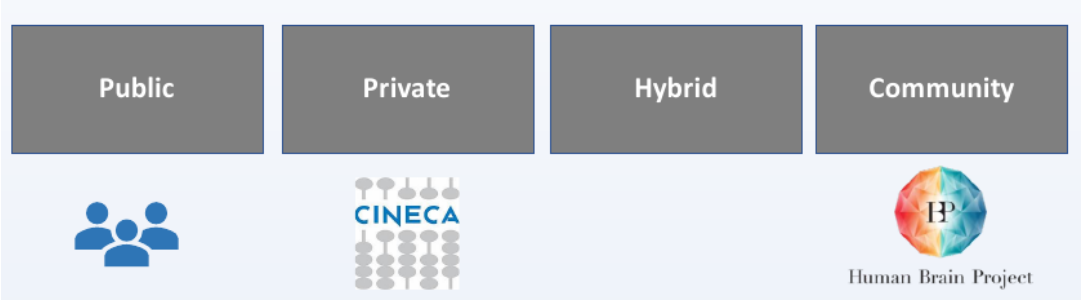
- 71 interactive OpenStack nodes each 2 x CPU Intel CascadeLake 8260, with 24 cores each, 2,4 GHz, 768GB RAM and 2TB SSD storage.
- 1 PB Ceph storage raw dedicated (full NVMe/SSD)

This cloud infrastructure is tightly connected both to the LUSTRE storage of 20 PB raw capacity, and to the GSS storage of 6 PB seen by all other infrastructure. This setup enables the use of all available HPC systems (Tier-0 Marconi, Tier-1 Galileo100), addressing HPC workloads in conjunction with cloud resources.

Top

Cloud Model

From the user's perspective, ADA cloud can be seen as both a public cloud and a community cloud, with a federation of European data-centers providing features targeting specific scientific communities (i.e. the flagship Human Brain project). ADA cloud HPC infrastructure is a resource that CINECA already adopts in several internal projects and services. The deployment model is well represented by the picture below.

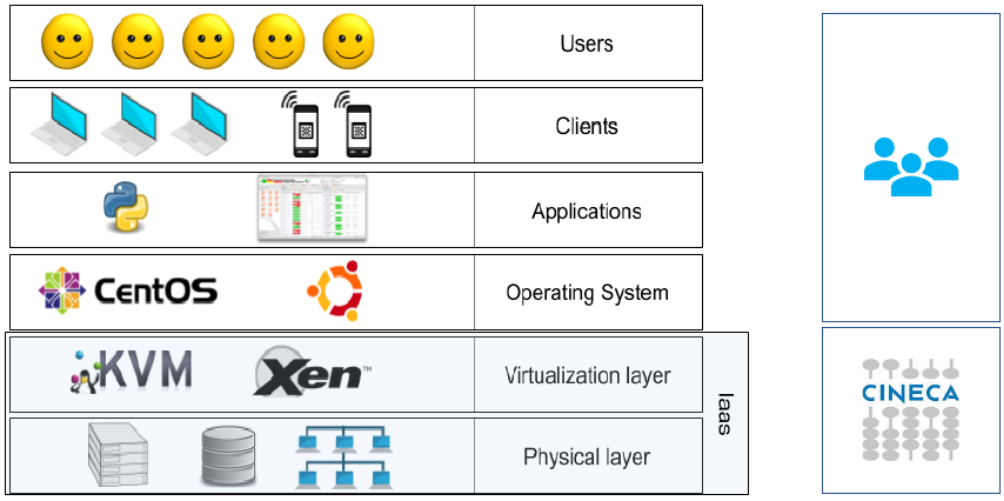


The ADA cloud HPC infrastructure integrates and completes the HPC ecosystem, providing a tightly-integrated infrastructure that covers both high performance and high flexible computing. We expect the flexibility of the cloud to better adapt to the diversity of user workloads, while still providing high-end computing power. If the need for High-Performance Computing increases, or scale beyond the ADA cloud HPC provision, the other world-class HPC systems (MARCONI, MARCONI100, GALILEO100) can be integrated into the workflow to cover all computing needs. For example, data can be stored on areas ($DRES) that are seen by all HPC systems.

Top

Service model

ADA cloud HPC infrastructure provides users an **Infrastructure as a Service** (IaaS). Along with all the advantages in terms of flexibility, there is an increased responsibility shifted from CINECA staff to users. A clear separation of roles in using the service is represented in the scheme below. This has to be understood by all actors accessing the service, even if we can provide assistance and share our expertise to help you set-up your application workflow.



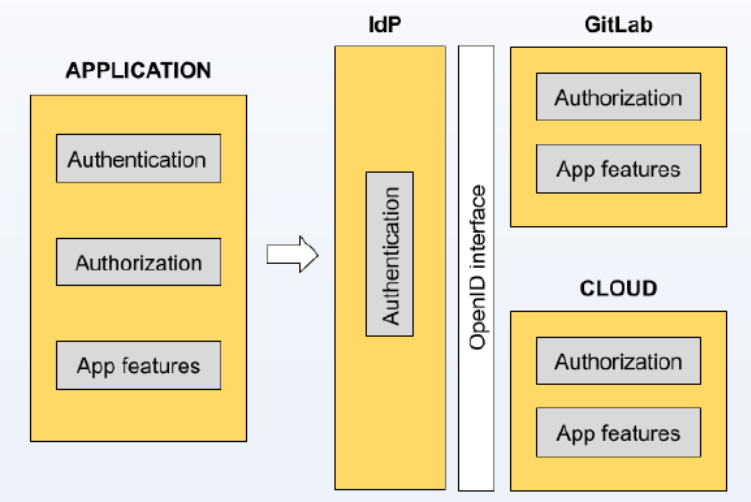
There are clear benefits in using a CLOUD infrastructure with access to **Virtual Machines** (VMs) with respect to our traditional HPC resources. These benefits can be summarized in the table below:

	Traditional HPC resources	Virtual Machines
Performance	Target the highest possible	depend on workload, but generally, virtualization has a small impact
User access	CINECA staff authorization	Once a project is granted, it is managed by the user
Operating System	It is chosen by CINECA staff given the HW constraints. Security updates are managed by CINECA.	Selected by the user. Security patch and updates are managed by the user.
Software stack	Mostly installed by CINECA staff. Users can install their own without "root" privilege. The environment is provided "*as is*"	The user is root on the VMs and can install all the required software stack. Users can modify the environment to suit their needs.
Snapshots of the environment	Cannot be done	User can save snapshot images of the VMs
Running simulations	Users are provided with a job scheduler (SLURM)	Users can install a job scheduler or chose alternatives.

Top

Flexible authentication model

A more flexible authentication method has been deployed in the CLOUD.HPC instance. It is based on OpenID (https://openid.net/connect/), and decouples authentication (access with credentials) from authorization (application permissions after user access), as represented in the schema below.

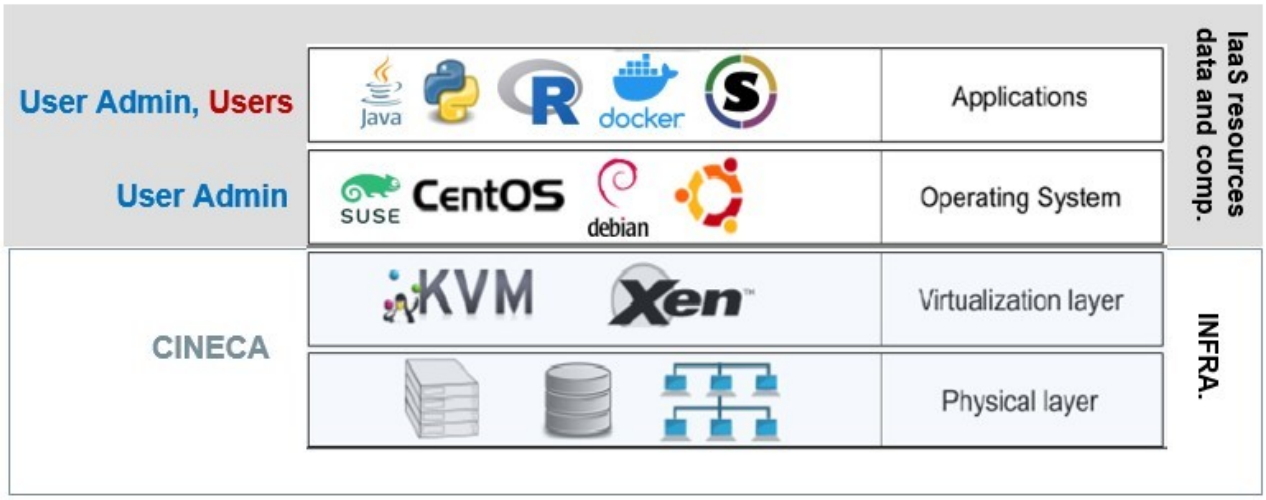


The Identity provider (IdP) can be internal (CINECA) or can be another trusted external service provider. This approach allows having in place federated identity, with a central (proxy) IdP servicing federated data-centers, as in the ICEI-Fenix model (https://fenix-ri.eu/).

Top

Roles and responsibilities

In the context cloud HPC resources provisioning, CINECA acts accordingly to the following division of roles:



- CINECA is responsible for administering the physical infrastructure and providing the virtualization layer (via Openstack)
- “User Admins” and “Users” are roles acted by people external to CINECA staff (Exceptions are made for internal services). User Admins can creat e VM instances and configure the resources via dashboard; “Users” do not access the dashboard and are local to each VM instance (for example those added via add user linux command).

Any user (“User Admins" or “Users”) with administration privileges on IaaS resources (VMs) have the responsibility to maintain the security (security patch, fix) on those resources. In particular, he/she has the responsibility to perform VMs and volume data backup also.

From the project management perspective, CINECA will interact only with “User Admins" (User Admins are user associated to the project in CINECA resource provisioning portal, https://userdb.hpc.cineca.it).

At the end of the project validity, the “User Admins” will receive communication from CINECA staff that the project as expired with the date by when the resources will be removed. It is “User Admins” responsibility to make copy of the necessary VMs or data before that date.

Top

Quotas, Flavors and Images

Quotas

Each project is assigned a quota that defines the resources it can use.

When resource consuming operations such as virtual machine creation are performed, the request is validated against the maximum quota permitted for the current project (as set by the environment variables or Horizon dashboard).

The default project quota is:

Compute	Limit		Volume	Limit
Instances	10		Volume Storage	10
VCPUs	4		Volume Snapshots	10
RAM (GB)	30		Volume Storage (GB)	512

Network	Limit
Floating IPs	1
Security Groups	10
Security Group Rules	20
Networks	2
Ports	50
Routers	2

Flavours

A flavor defines the virtual machine size such as

- Number of VCPUs
- Amount of memory
- Disk space (system disk, ephemeral disk, and swap)

A standard set of flavors allows predictable distribution of applications across multiple hypervisors.

Flavour Name	VCPUs	RAM GB	Disk GB	Public-available for all projects
fl.ada.xxs	1	7,5	10	yes
fl.ada.xs	2	15	30	yes
fl.ada.s	4	30	30	yes
fl.ada.m	8	60	30	yes
fl.ada.l	16	120	30	yes
fl.ada.xl	24	180	30	On-demand
fl.ada.xxl	48	360	30	On-demand
fl.ada.full	96	720	30	On-demand

Images

OpenStack images provide the source for booting a virtual machine. An image consists of an operating system, some optional additional packages and some flags to help OpenStack place these on the right hypervisor.

For more detailed information about OpenStack's image management, the OpenStack image creation documentation provides further references and links.

The complete list of provided images is available by click on the tab "Images" and then on "Public" tab on the right in the Openstack Dashbord.

In what follow there is the list of the default images provided for all projects.

IMPORTANT NOTE: It is not admitted building Windows virtual machine on ADA cloud, even if the user has its own windows license.

Image Name	Image information	Default user	Password
CentOS-7-x86_64-GenericCloud-2009	CentOS-7-x86_64-GenericCloud-2009.qcow2, last modified 2020-11-12 Source: http://cloud.centos.org/centos/7/images/	centos	ssh access by key
CentOS-8-GenericCloud-8.4.2105-20210603.0.x86_64	CentOS-8-GenericCloud-8.4.2105-20210603.0.x86_64, last modified 2021-06-03 Source: https://cloud.centos.org/centos/8/	centos	ssh access by key
Ubuntu 18.04 LTS (Bionic Beaver)	Ubuntu server 18.04 (Bionic Beaver) LTS for cloud Source: https://cloud-images.ubuntu.com/	ubuntu	ssh access by key
Ubuntu Server 20.04 LTS (Focal Fossa)	focal-server-cloudimg-amd64.img, last modified 2021-07-20 Source: https://cloud-images.ubuntu.com/	ubuntu	ssh access by key
Ubuntu Server 21.04 (Hirsute Hippo)	hirsute-server-cloudimg-amd64.img, last modified 2021-07-20 Source: https://cloud-images.ubuntu.com/	ubuntu	ssh access by key
Ubuntu Server 22.04 LTS (Jammy Jellyfish)	jammy-server-cloudimg-amd64.img, last modified 2022-09-02 Source: https://cloud-images.ubuntu.com/	ubuntu	ssh access by key
Rocky Linux 8.9	File description: https://wiki.rockylinux.org/rocky/image/#about-cloud-images Source: https://dl.rockylinux.org/pub/rocky/8/images/x86_64/Rocky-8-GenericCloud-Base-8.9-20231119.0.x86_64.qcow2	rocky	ssh access by key
Rocky Linux 9.3	File description: https://wiki.rockylinux.org/rocky/image/#about-cloud-images Source: https://dl.rockylinux.org/pub/rocky/9/images/x86_64/Rocky-9-GenericCloud-Base-9.3-20231113.0.x86_64.qcow2	rocky	ssh access by key

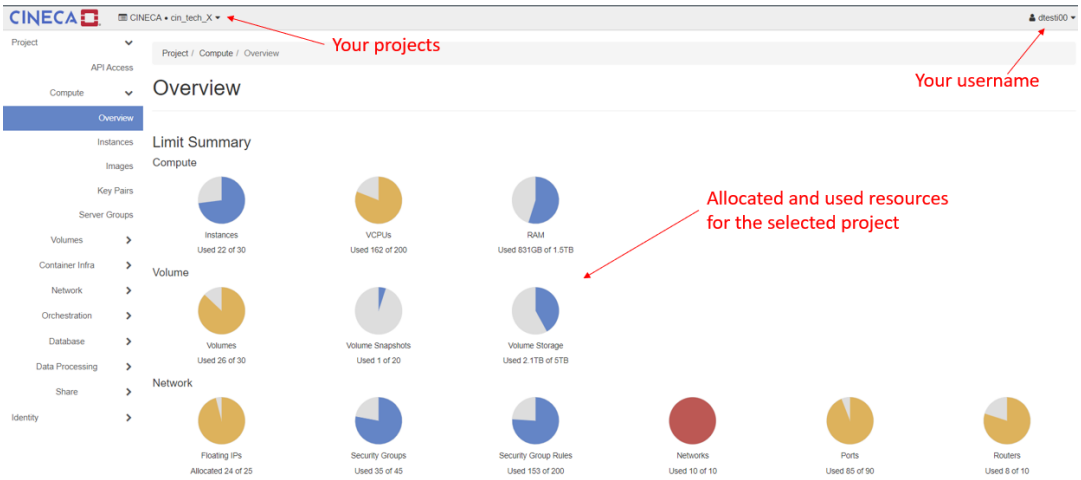
Top

Access

Log in to the OpenStack dashboard

Go to the OpenStack dashboard at https://adacloud.hpc.cineca.it, select "CINECA HPC" as Authentication method, then insert your HPC-CINECA credentials to log in, together with the 2nd factor for authentication (see section How to connect via 2FA for more information).

After the log in, on the top-right of the window is displayed your user name, while on the top-left. are listed in a menu all the Projects you are associated with.



Projects are organizational units in the cloud. Each user is a member of one or more projects. Within a Project, a user can create and manage instances, security groups, volumes, images, and more.

From the Project tab, you can view and manage the resources assigned in a particular project, including instances, images and volumes. Moreover you can select a period of interest and have the usage summary.

You can select one of the project you are associated with the menu on the top-left side of the window.

Log in to the virtual machine

After you have created your virtual machine (see the following section) you can log in directly using:

- the default user and key (if you have used a native default image for cloud)

- another username (if you have used your personal image with a custom user defined in it)

Suppose you have used the default Ubuntu cloud image, you can login as:

```
$ ssh -i MyKey.pem ubuntu@<floating IP address>
```

Top

Creation of an instance of the virtual machine in your Project

In order to create your own virtual machine you have to perform all the following eight steps

1. Log in to the dashboard https://adacloud.hpc.cineca.it as described in the Access section.

2. Check and **configure** the Internal Network in the Project

In order to build and use virtual machine within a specific Project, it is mandatory the presence of the internal network, subnet and router.

Select the Project of interest and check the presence of such components click on tab Project Network Network Topology.

If it is present only the "external network", you must create network, subnet and router. Please, follow the instruction below:

2.1 Create a private network and subnet
To create a private network and subnet click on: Project -> Network -> Network Topology -> Create Network
In case you do not know how to configure a Network, here below default parameters for the configuration are provided.

- Tab Network

  Network name: <the name you want>

  Enable Admin State: check

  Create Subnet: check

  Availability Zone Hints: set "nova"

  MTU: leave it blank (this will correspond to a default of 1450)

- Tab subnet

  Subnet name: <the name you want>

  Network Address: 192.168.0.0/24

  IP Version: set IPv4

  Gateway IP: 192.168.0.254 (the last address for subnet 192.168.0.0/24)

  Disable Gateway: disabled (uncheck)

- Tab Subnet Details

  Enable DHCP: enabled (check)

  Allocation Pools: leave blank

  Host Routers: leave blank

- Finally, click on "create"

2.2 Create a private router and set the gateway

Click on: Project -> Network -> Routers -> Create Router

- Then set:

  Router name: <the name you want>

  Enable Admin State: check

  External Network: select "externalNetwork"

  Availability Zone Hints: leave "nova"

- Finally, click on "create router".
- Now, select the router just created and click on "Interfaces" and then on "Add interface"

  subnet: select the subnet just created

  IP address: 192.168.0.254 (this has to be the same IP address of the gateway)

- Finally, click on "Submit".

Verify, in the Routers/Overview tab, that the Status of router is “ACTIVE” and the Admin state is “UP”.

3. Set up a keypair

Keypairs are used to access virtual machines when:

a. the instance is launched using a default image for cloud (e.g. centos or ubuntu)
b. in the virtual machine is set a login with ssh -key

You can set up a keypair in two ways. From "Project Compute Key Pairs" menu, you can:

- click on "Create Key Pair", to obtain a new key pair. The possible types are SSH key or x509.
- click on "import Public Key" to import your key pair.

When you click create, the public key stays on the Openstack dashboard, the private key is downloaded locally. The download of the private key will be done only when the keypair is created. It will not be possible to re-download it. If you lose the private key you will have to create a new keypair.

NOTE: if you are a Linux user, remember to modify the permission of the private key (downloaded file) to read-write for only the user (chmod 600 <file name>) in order to avoid errors when you use it to login to your virtual machine.

4. Set the security rules, that will be the firewall of your virtual machine

The firewall of the virtual machine must be defined using the OpenStack Security Groups and Security Rules.

- A security rule defines which traffic is allowed to instances assigned to the security group.
- A security group is a group of security rules that can be assigned to an instance.

Inside the virtual machine, the firewall must be disabled.

The security groups and security rules can be created click on "Project Network Security Groups"

- at the start only the "default" security group is present (which does not have any ingress rule set up)
- click on "create security group"

At this point, by default the just created security group only has the rules for the outgoing traffic of the machine; you need to add the rules for the incoming traffic.

Common default rules are:

- SSH (port 22)
- ICMP (allow to "ping" a server)
- HTTP (port 80)
- HTTPS (port 443)

Note: It is always possible to modify, add and remove security groups in a virtual machine after its creation.

- If you modify a security group, adding or removing rules, and the security group is already associated to the virtual machine, the changes will be available in real time
- If you want to add or remove a security group from a virtual machine, click on "Project Compute Instances", select the virtual machine and from the menu on the right, click on "Edit Security Group". So, add or remove the security groups to the instance.

5. Launch an instance of Linux virtual machine

Once your key pair and your security group are defined, proceed building the virtual machine.

- Click on "Project Compute Instances"
- Click on "Launch instance" button
- In the "Details" box, enter:
  - the instance name
  - number of instances you want to create with this configuration (default is 1)
- In the "Source" box, enter:
  - the boot source for the instance. It can be an image, a bootable volume or a bootable volume snapshot.
    - Images: we provide some default images (centos, ubuntu, etc.). For these default images, it is set a default user can login into the virtual machine using a key pair. Such a user can execute commands as root. The password of the user root is embedded. If you want to use your personal image, you can create it in the cloud environment click on "Project Compute Images", the "Create Image" and upload it.
    - Note If you want to create a bootable volume from your instance, select "yes" in "Create New Volume" and select the size of such volume.
- In the "Flavor" box, select the flavor you want to use, accordingly with the resources you have.
    - NB: if you select to create a volume from your instance, the root disc of the virtual machine will have the size of the volume, not the size set in the flavor
- In the "Networks" box, enter the network internal to your project on which connect the virtual machine
- In the "Security Groups" box, select the security groups you want. Remember that you can always modify them after the virtual machine creation.
- In the "Key Pair" box, select the key pair you want to use for ssh login.

6. Follow the **boot process**

The boot process can be followed on the instances screen. Once the VM is in state ACTIVE, you will be able to open the console and follow the boot process.

To follow the installation, you can access the graphical console using the browser once the VM is in BUILD state.

The console is accessed by selecting the "Instance Details" for the machine and then click on the tab "Console".

7. Associate a Floating IP (FIP) to the virtual machine

Where floating IPs are configured in a deployment, each project will have a limited number of floating IPs controlled by a quota. However these need to be allocated to the project from the central pool prior to their use.

To allocate a floating IP to a project, click on "Project Network Floating IPs", then click on the button "Allocate IP to project" on the right side of the dashboard page. Once allocated, a floating IP can be associated with running istances. Just click on "Associate" action on the right of the page. In the popup, select your virtual machine by the menu in "Port to be associated".
The inverse action, Dissociate Floating IP, is available from the "Instances" page.

8. Login to the virtual machine using ssh

After the association of a Floating IP to your virtual machine, you can login using the default user and key (if you have used a native default image for cloud), or using another username (if you have used your personal image with a custom user defined in it). Suppose you have used the default ubuntu cloud image, you can login as:

$ ssh -i MyKey.pem ubuntu@<floating IP address>

Top

Operations with an instance

Create an instance snapshot

1. Log in to the virtual machine and shutdown the instance
2. Log in to the dashboard, choose the right project, and click *Instances*.
3. Check that the instance to snapshot is Shutoff
4. Select such instance
5. In the *Actions* column, click *Create Snapshot*.
6. In the *Create Snapshot* dialog box, enter a name for the snapshot, and click *Create Snapshot*.The Images category shows the instance snapshot.

To launch an instance from the snapshot, select the snapshot and click *Launch*. Proceed with launching an instance.

Manage an instance

1. Log in to the virtual machine and shutdown the instance
2. Log in to the dashboard, choose the right project, and click *Instances*.
3. Select an instance.
4. In the *More* list in the *Actions* column, select the state.

   You can resize or rebuild an instance. You can also choose to view the instance console log, edit instance or the security groups. Depending on the current state of the instance, you can pause, resume, suspend, soft or hard reboot, or terminate it.

Track usage for instances

You can track usage for instances for each project. You can track costs per month by showing metrics like number of vCPUs, disks, RAM, and uptime for all your instances.

1. Log in to the dashboard, choose a project, and click *Overview*.
2. To query the instance usage for a month, select a month and click *Submit*.
3. To download a summary, click *Download CSV Summary*.

Monitor instances

You can monitor the high-level actions (creation, start, stop) on the instances for each project via offered logs in the dashboard.

1. Log in to the dashboard, choose a project, and click *Instances*.
2. To monitor the logs of the instance usage for a month, select the instance of your interest.
3. Go to the *Action log*

More detailed monitoring logs can be set-up by you within the specific instance.

Rescue a Virtual Machine

Instance rescue provides a mechanism for access, even if an image renders the instance inaccessible. Two rescue modes are currently provided.

Case: Ephemeral Virtual Machine

IMPORTANT: If the virtual machine has encrypted **LUKS VOLUMES** attached, it is mandatory to detach them before starting the rescue operation.

The steps needed to rescue an inaccessible ephemeral virtual machine on ADA Cloud are:

Step 0.

- Create a **rescuer** virtual machine with a **new key pair**. Although this is not a fixed rule, it is suggested to create the rescuer machine using an image with **same OS** as the one on the inaccessible machine (same version or newer).
- Login to the rescuer and update it. As an example, for Ubuntu virtual machines:

```
sudo apt update
sudo apt upgrade
```

- Logout the rescuer and create a **snapshot image** of this virtual machine.

Step 1.

- Select the instance you want to rescue and from the drop-down menu on the right select "**rescue instance**":
  - In the menu that appears, select the image you just created from the rescuer machine.
- Login via ssh to the broken machine using the rescuer username/key
- Check that the boot of the machine has been correctly executed using the command

```
lsblk
```

  you should see the **rescuer** machine (**/dev/vda1**) mounted and the inaccessible machine on the device **/dev/vdb1**
- Mount such device /dev/vdb1

```
sudo mkdir /mnt/inaccessible_vm
sudo mount /dev/sdb1 /mnt/inaccessible_vm
```

- Now you can access the files in the inaccessible machine to fix the problems (lsblk, fsck, xfs_repair, chroot, etc.) or backup important data
- Once the operation is done, logout the virtual machine and from the Dashboard select "**unrescue**".

Case: Virtual Machine booted from a Bootable Volume

IMPORTANT: If the virtual machine has encrypted **LUKS VOLUMES** attached, it is mandatory to detach them before starting the rescue operation.

The steps needed to rescue an inaccessible VM instantiated from a bootable volume on ADA Cloud are:

Step 0.

- Shutdown the instance.
- In the tab "Volumes", track which secondary volumes are attached to the VM to be rescued and detach them.
- **IMPORTANT**: verify that the bootable volume won't be erased when deleting the VM!
  - To do this check, execute the command "openstack server volume list <id-vm-to-be-rescued>", have a look to the field "delete_on_termination?" that must be set to 'False'. (Note that this will work with openstack-cli >= 6.2.0)
  - NB: In case you don't have access to the OpenStack CLI, please contact superc@cineca.it
- Keep track of the Flavor, Security Groups and FIP associated with the VM (FIP in particular if there is a DNS association).
- Delete the instance.

Step 1.

- Create a throwaway VM, attach the bootable volume to rescue as a secondary volume and associate a FIP to such VM.
- Login via ssh to the throwaway VM and execute all the needed operations on the volume to rescue (lsblk, fsck, xfs_repair, chroot, etc.).
- Once the volume has been recovered, exit the throwaway VM and detach the secondary volume that has been rescued.
- Restart the VM from the rescued bootable volume, reattaching the secondary volumes, FIP, and check the problem has been solved.

Resize a VM (a.k.a. change its flavor)

Users are able to resize autonomously their VM, but, since there is a **bug in current version the OpenStack Horizon in ADA cloud,** this operation can be done only **via OpenStack CLI**.

To know how to configure and use the OpenStack CLI, please refer to the link OpenStack CLI .

Before to perform the resize operation:

1. The VM must be shut off.
2. If there are encrypted **LUKS VOLUMES** attached to the virtual machine, it is mandatory that the user:
   - **Unmount** the volumes from the VM
   - **Detach** the volumes from the **Openstack Dashboard**

To resize a VM, it is necessary to:

- Identify the VM ID:

```
openstack server list --all | grep <VM_name>
openstack server show < vm_ID > | grep flavor
```

- Identify the ID of the new flavor the VM needs:

```
openstack flavor list
```

    - In the case of an Ephemeral VM, check the size of root disk of the original VM. Don't resize the VM if the new flavor has a disk smaller than the current one.
    - In the case of a VM with a Bootable Disk, the resize will affect only VCPUs number and RAM. The bootable disk will not be changed by the operation.

- Perform the resize, remembering to alert the user of the VM's temporary shutdown during the operation.

```
openstack server resize --flavor <new_flavor_ID> --wait <vm_ID>
```

- Wait then the operation to "**Complete**"; at the same time, on the Openstack Dashboard the message "**Confirm**" will appear next to the server name. Then use the command:

```
openstack server resize confirm <vm_ID>
```

  Issue the resize confirmation in a separate command, since the option `--confirm` on the command `openstack server resize` is deprecated.

- Verify the success of the operation. Since the Dashboard can have visualization bugs, it is best to check via CLI:

```
openstack server show < vm_ID > | grep flavor
```

- Ask the user to confirm the success of the operation. To do that they will need to boot the VM, login, and verify the VCPUs number and Memory size are correct with the following commands:

```
cat /proc/cpuinfo
free -g
```

Resize of a VM's Bootable Volume

If a user owning a VM with a Bootable Volume needs to resize it to make it bigger (never SMALLER, in order to avoid breaking the VM), they can perform the operation themselves via Openstack Dashboard:

- Checking there is enough free space usable on their tenant;
- Shutting down the VM;
- Clicking on the Bootable Volume, then on Extend Volume inside the right menu, writing there the desired new volume size;
- Rebooting the VM and check that the volume has the correct size, using the command `df -h` .

Download a VM

To download a VM, it is necessary to create a **snapshot** of it on the **Openstack Dashboard**, and then **save locally** the snapshot using the **Command-line** interface.

- **Shutdown** the VM
- **Detach** any secondary volume attached on the VM (remember the volume `/dev/vda` is the bootable volume from which the VM is loaded).
- **Create** the VM snapshot
    - If the VM is **loaded from an image** (there is no bootable volume `/dev/vda`):
        - Click on **create snapshot**, and select format **qcow2**.
        - The snapshot should appear in the image list with size different from zero.

    - If the VM is **loaded form a bootable volume** (volume attached as `/dev/vda`):
        - Select the bootable volume,
        - Create a snapshot of the volume,
        - Create a volume from this snapshot.
        - From the new created volume, click on **upload to image** and then choose as disk format **qcow2 .** (this operation is slower than the other)
        - The snapshot should appear in the image list with size different from zero.
- Now from the **CLI**:
    - **Setup** the environment
    - Search the snapshot image created earlier, and get the **Image ID**

```
openstack image list
```

    - Save the image on local hardware

```
openstack image save --file <img_file_name> <image_snapshot_ID>
```

    - Check the downloaded image info to be sure the process has been executed correctly

```
qemu-img info <img_file_name>
```

Top

Cinder Volumes

Volumes are block storage devices that you attach to instances to enable persistent storage.

After the creation, you must attach the volume to a running instance and then mount it from inside the virtual machine. A volume can be also detached from an instance, and attached to another instance at any time. It is also possible to create a snapshot from a volume and/ or delete it.

Operations with Cinder Volumes

1. **Creation of a Cinder Volume**

   a) Log in to the dashboard, choose a project, and click on "Projects Volumes Volumes".

   b) Click on "Create Volume" button:

   In the dialog box that opens, enter or select the following values.

   Volume Name: Specify a name for the volume.

   Description: Optionally, provide a brief description for the volume.

   Volume Source: Select one of the following options:

   - No source, empty volume: Creates an empty volume. An empty volume does not contain a file system or a partition table.
   - Image: If you choose this option, a new field for Use image as a source displays. You can select the image from the list.

   Type:

   - __DEFAULT __ is a general cinder volume
   - LUKS is for encrypted volumes.

   Size (GiB): The size of the volume in gibibytes (GiB).

   c) Finally, click on Create Volume button.

The dashboard shows the volume on the Volumes tab.

1. **Attach the volume to a running instance**

   After you have created the volume, you have to attach it to instances, in order to use it. You can attach a volume to one instance at any time.

   a. Log in to the dashboard, choose a project, and click on "Projects Volumes Volumes".
   b. Select the volume to add to an instance and click "Edit Attachments".
   c. In the *Manage Volume Attachments* dialog box, select an instance.
   d. Enter the name of the device from which the volume is accessible by the instance.
   e. Click *Attach Volume*.

      The dashboard shows the instance to which the volume is now attached and the device name.

   You can view the status of a volume in the Volumes tab of the dashboard. The volume is either Available or In-Use.

   Now you can log in to the instance and mount, format, and use the disk.

2. **Format and mount a volume attached to an instance**

After a cinder volume has been created and attached to a virtual machine using the OpenStack, in order to use for storing data its needed to partition, format and mount it. This operations has to be done inside the virtual machine.

Following, some suggestion to perform these operations

Partition Table

Suppose that the volume is attached to the virtual machine as device /dev/vdc.

Login in to the virtual machine and use fdisk to modify the partition table.

- list the partition table

sudo fdisk -l

- partition of device /dev/vdc

sudo fdisk /dev/vdc

1 new partition, primary, with default sector numbers and type "Linux"

==> n; p; 1 ; ...default ;

check and write

==> p; w

Format the device /dev/vdc just partitioned as xfs:

sudo mkfs -t xfs /dev/vdc1

Mount the volume:

sudo mkdir /mnt/stuff_1

sudo mount /dev/vdc1 /mnt/stuff_1

To mount the volume automatically at each boot of the virtual machine, please modify the /etc/fstab file.

Following the example, in the /etc/fstab could be written:

/dev/vdc1 /mnt/stuff_1 xfs auto,nofail,defaults 0 0

4. **Detach a volume from an instance**

   a. Log into the virtual machine and umount the volume.

      Take care: if you had written the automatic mount of the volume in the file /etc/fstab, please comment or cancel the line in the file.
   b. Log in to the dashboard, choose a project, and click **Volumes**.

c. Select the volume and click **Edit Attachments**.
d. Click **Detach Volume** and confirm your changes.

A message indicates whether the action was successful.

5. **Create a snapshot from a volume**

a. Log in to the dashboard, choose a project, and click *Volumes*.
b. Select a volume from which to create a snapshot.
c. From the *More* list, select *Create Snapshot*.
d. In the dialog box that opens, enter a snapshot name and a brief description.
e. Confirm your changes.

The dashboard shows the new volume snapshot in Volume Snapshots tab.

6. **Delete a volume**

When you delete an instance, the data in its attached volumes is not destroyed.

a. Log in to the dashboard, choose a project, and click *Volumes*.
b. Select the check boxes for the volumes that you want to delete.
c. Click *Delete Volumes* and confirm your choice. A message indicates whether the action was successful.

Top

Sharing a filesystem among virtual machines: the Manila service

An user can create volumes ("shares") that are shared among virtual machines. This is provided by the Manila service inside Openstack.
Instruction to be followed to create this storage can be found in this dedicated page.

Top

Storing of sensitive data

Sensitive data can be stored on special encrypted Cinder Volume of type LUKS.

By using the Openstack dashboard, every user can create such volumes and then attach them to a virtual machine. Due to a limitation of the crypto library, the maximum size of each volume is 15 TB.

Since LUKS are encrypted volumes, the time needed to create one can vary greatly with respect to the size of the volume to create (most of which is the time needed to encrypt the data).

Here are some indicative times needed to create different sized LUKS volumes from the dashboard (stime):

- 1 TiB: 15 minutes
- 7 TiB: 2 hours
- 10 TiB: 3-4 hours

The user can access the data stored in such LUKS volumes by login into the corresponding virtual machine. Only the users with authorisation to login into the virtual machine will access the data "in clear", even if it is encrypted by key.

The keys used by the Openstack volume encryption feature are managed by Barbican, the official OpenStack Key Manager service. Barbican provides secure storage, provisioning and management of secret data. This includes keying material such as Symmetric Keys, Asymmetric Keys, Certificates and raw binary data.
