How to deploy containers on INFN-CLOUD

Corso base su *Docker* - September 12-14 2023
Marica Antonacci (INFN BA)

What is INFN-Cloud?

INFN Cloud is an internal project which aims to

- manage a (large) fraction of the INFN resources in a sustainable and optimized way;
- make different INFN communities able to access resources, regardless of the availability of local and dedicated hardware (including special hw like GPUs), of the availability of IT skilled people;
- focus on high-level added value services, not on “infrastructures”, to support:
  - ➢ Scientific Computing
  - ➢ Development and R&D, testing of new services
  - ➢ Training activities
  - ➢ Support to INFN data centers (for example for backups of services, etc)

INFN Cloud is built on top of INFN experiences, know-how and solutions developed during several projects and initiatives.

The INFN Cloud architecture

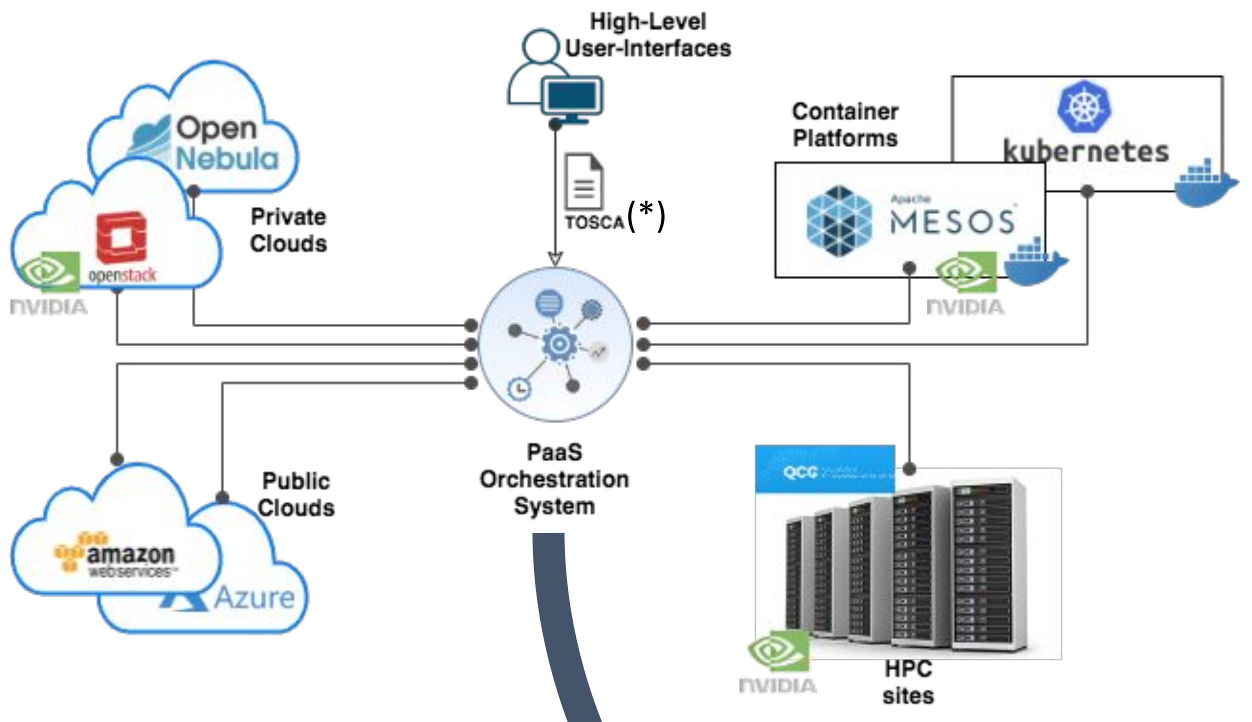
Architecturally INFN Cloud is a **federation** of existing infrastructures

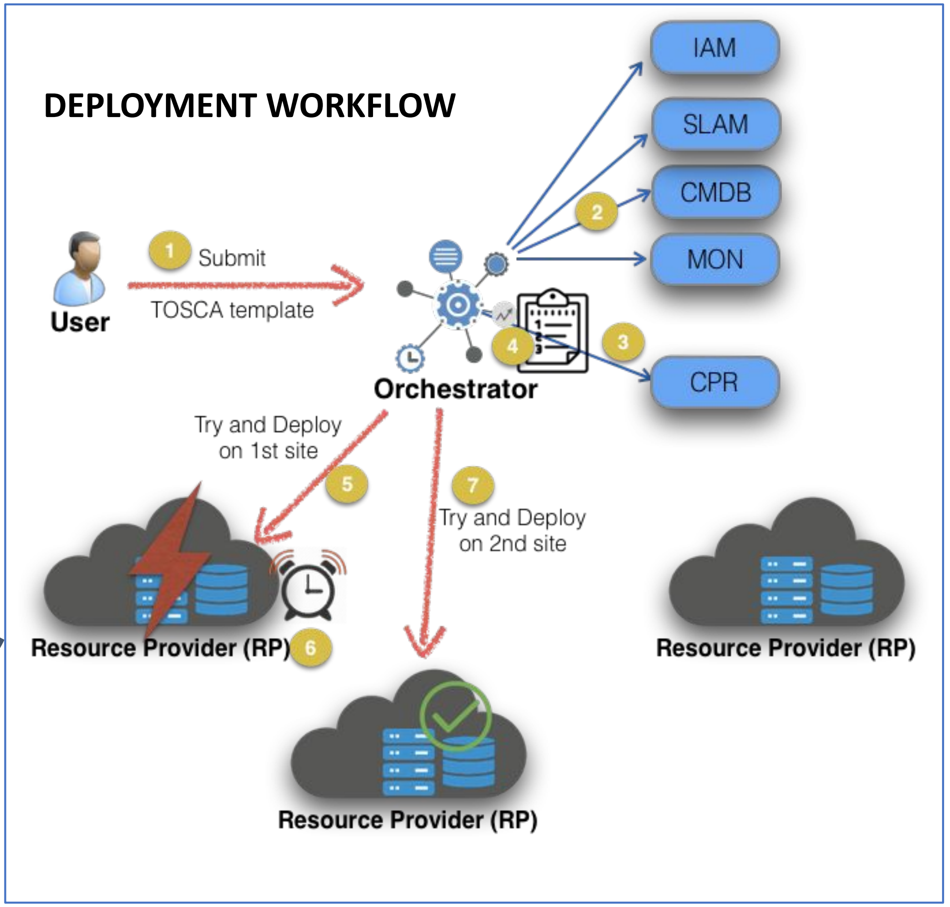
- ❑ **the INFN Cloud backbone**, that consists of two tightly coupled federated sites: BARI and CNAF
- ❑ **a scalable set of satellite sites**, geographically distributed across Italy, and loosely coupled.
  - Currently Cloud@CNAF, CloudVeneto and ReCaS-Bari are federated with the backbone

Key enabling factors for the federation:

- ❑ leverage the same authentication/authorization layer based on **INDIGO-IAM**
- ❑ agree on a consistent set of policies and **participation rules** (user management, SLA, security, etc.)
- ❑ transparent and dynamic orchestration of the resources across all the federated infrastructures through the **INDIGO PaaS Orchestrator**

PaaS Orchestration System (from 10Km)





(*) **T**opology and **O**rchestration **S**pecification for **C**loud **A**pplications

Ref: [TOSCA Simple Profile in YAML Version 1.1]

The INFN-Cloud services

Virtual Machines (VM) possibly with external volume for storing data.

Docker containers

Pre-configured environment for **data analytics**

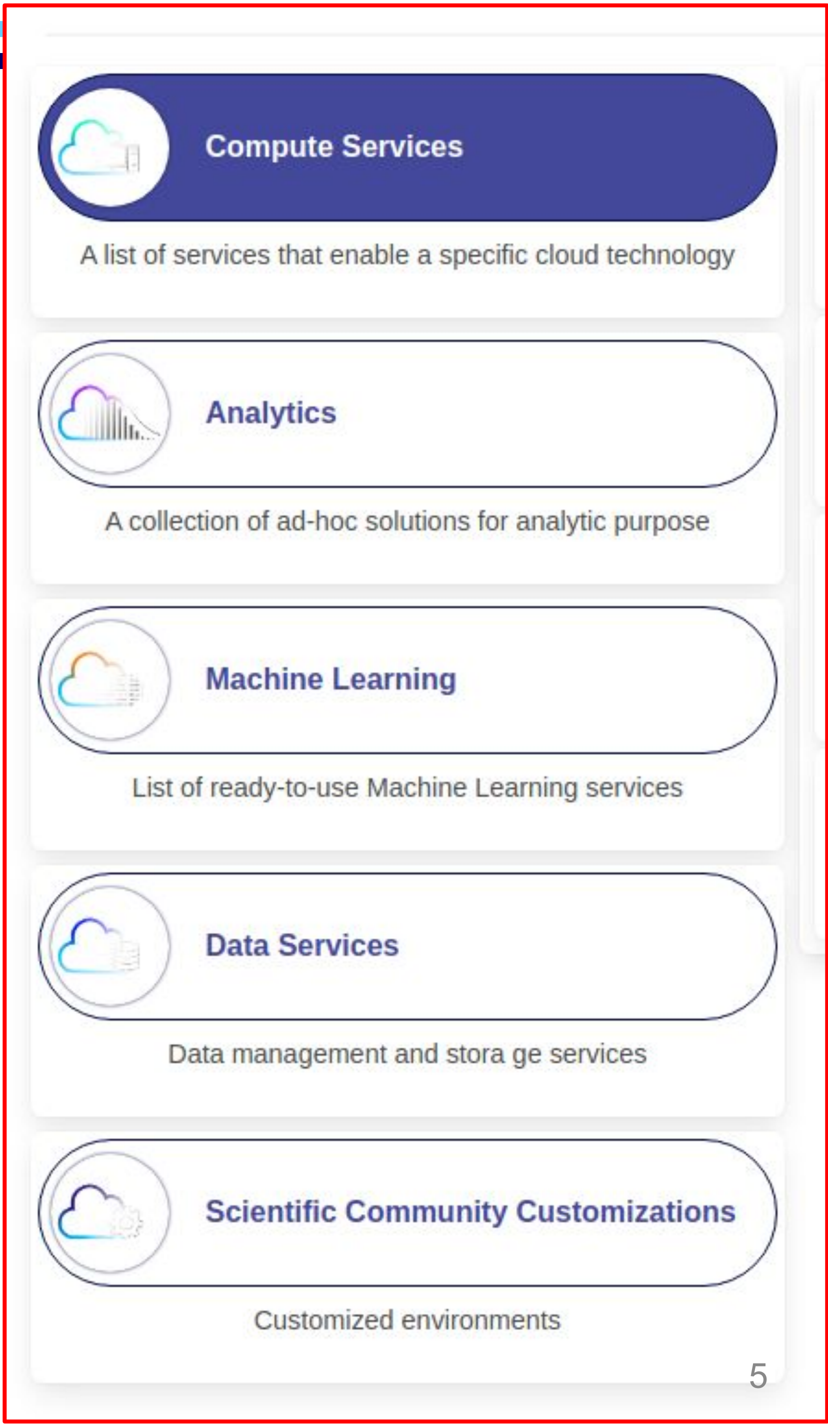
- Spark e/o ElasticSearch e Kibana, R, etc..

Storage solutions: Object storage/posix, possibly connected to high level application layers;

- Jupyter Notebooks with persistent storage (replicated)

Dynamic Clusters even designed and tuned taking into account the specific communities needs;

- HTCondor batch system; environment optimized for ML i.e. equipped with GPUs
- Container orchestrators such as K8s and Mesos


Compute Services

A list of services that enable a specific cloud technology

Analytics

A collection of ad-hoc solutions for analytic purpose

Machine Learning

List of ready-to-use Machine Learning services

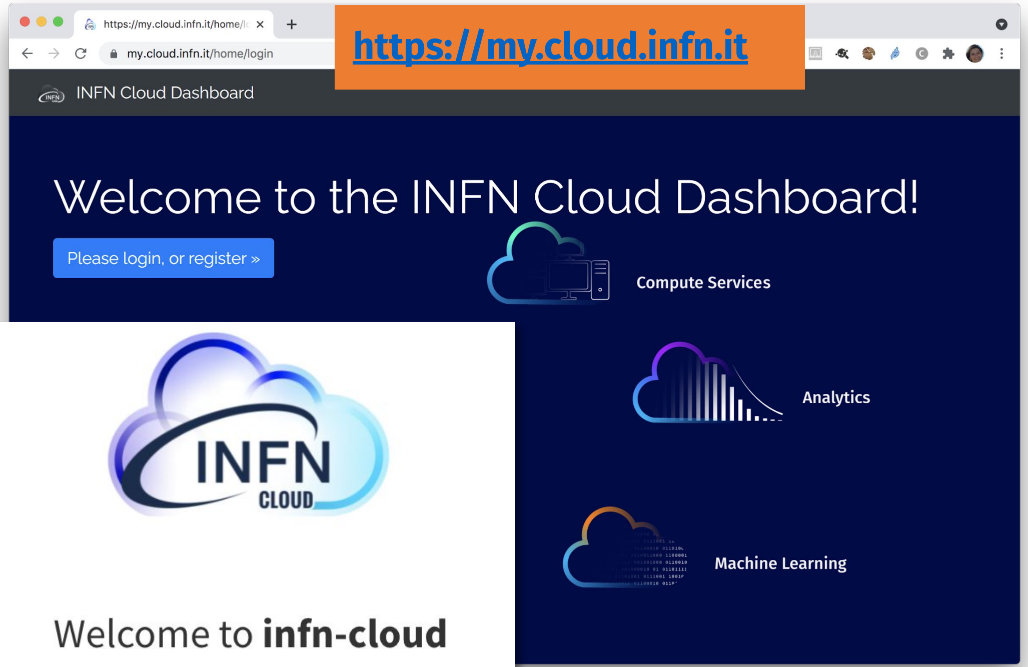
Data Services

Data management and stora ge services

Scientific Community Customizations

Customized environments

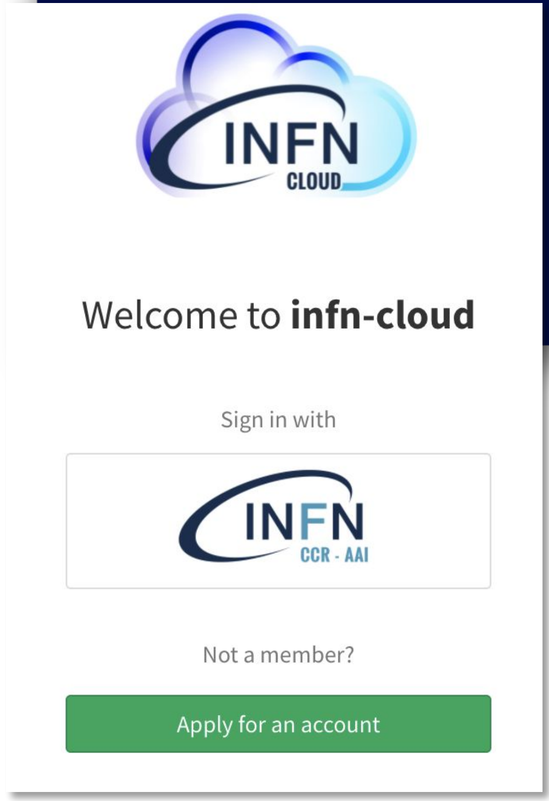
The INFN Cloud Dashboard


https://my.cloud.infn.it

INFN Cloud Dashboard

Welcome to the INFN Cloud Dashboard!

Please login, or register »

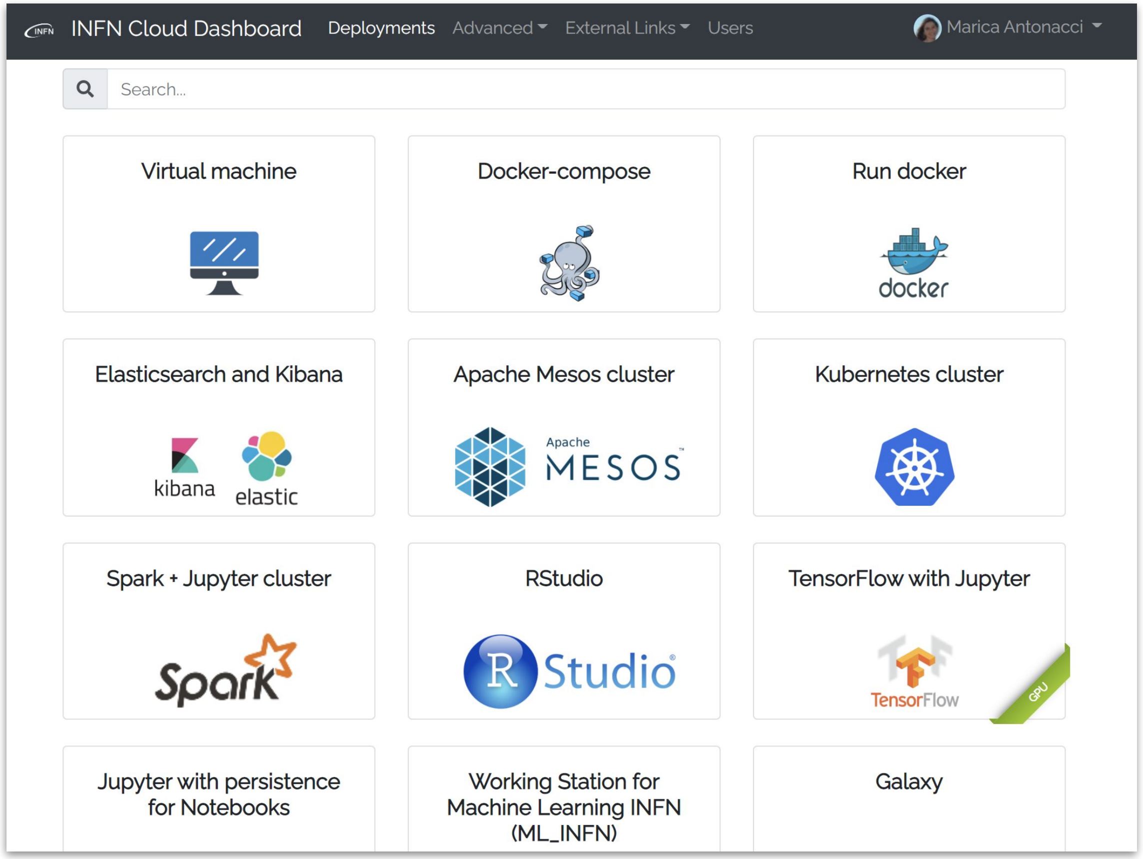

Welcome to **infn-cloud**

Sign in with

Not a member?

Apply for an account

INDIGO IAM manages the authentication/authorization through the whole stack (from PaaS to Iaas)



Users are organized in different IAM **groups**.

Each group can access a specific set of services from the dashboard (personalized view) and is mapped onto a dedicated tenant on the federated clouds.

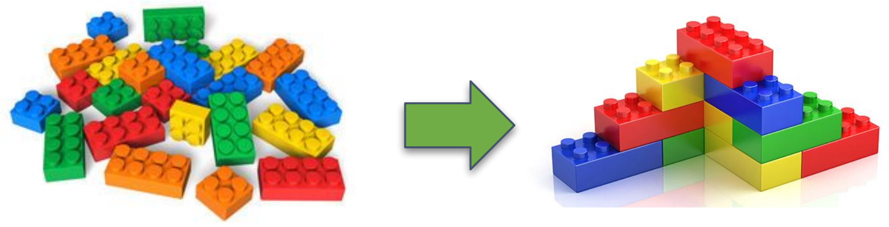
The service catalogue

The catalogue is a graphical representation of the TOSCA templates repository that we have been developing extending the INDIGO-DC custom types

- Each card in the catalogue is associated to one or more templates
- We are following a **lego-like** approach, building on top of reusable components and exploiting the TOSCA service composition pattern

Main objectives:

#1 - build added value services on top of IaaS and PaaS infrastructures
#2 - lower the entry barrier for non-skilled scientists

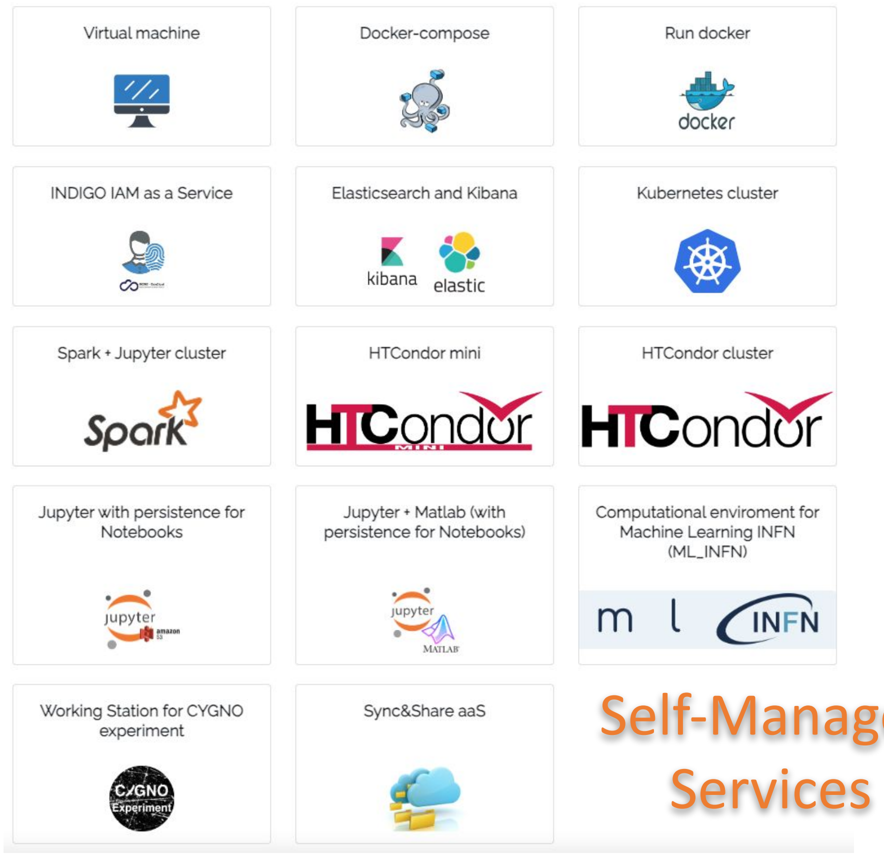


Available services

harbor.cloud.infn.it minio.cloud.infn.it

hub.cloud.infn.it

Fully-Managed Services



Virtual machine

Docker-compose

Run docker

INDIGO IAM as a Service

Elasticsearch and Kibana

Kubernetes cluster

Spark + Jupyter cluster

HTCondor mini

HTCondor cluster

Jupyter with persistence for Notebooks

Jupyter + Matlab (with persistence for Notebooks)

Computational enviroment for Machine Learning INFN (ML_INFN)

Working Station for CYGNO experiment

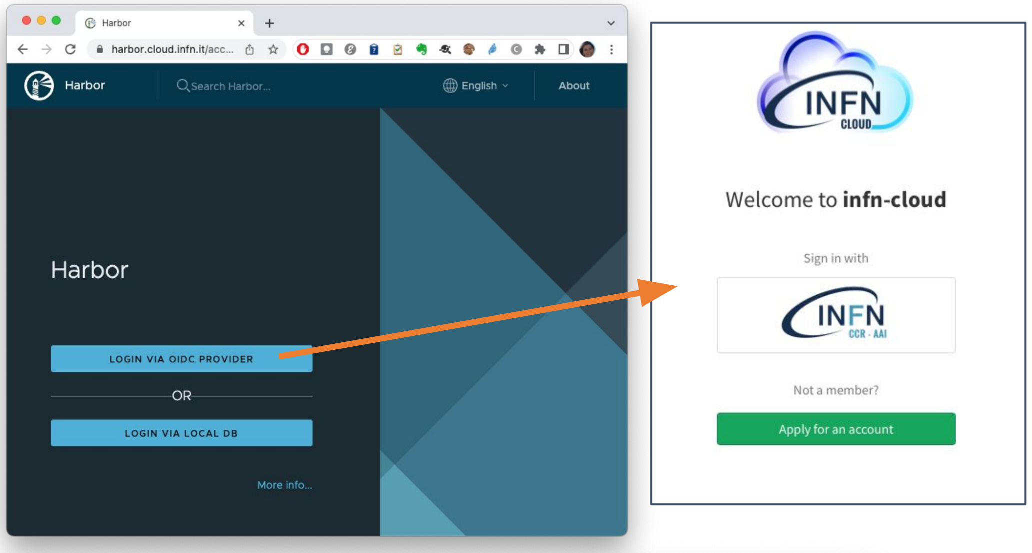
Sync&Share aaS

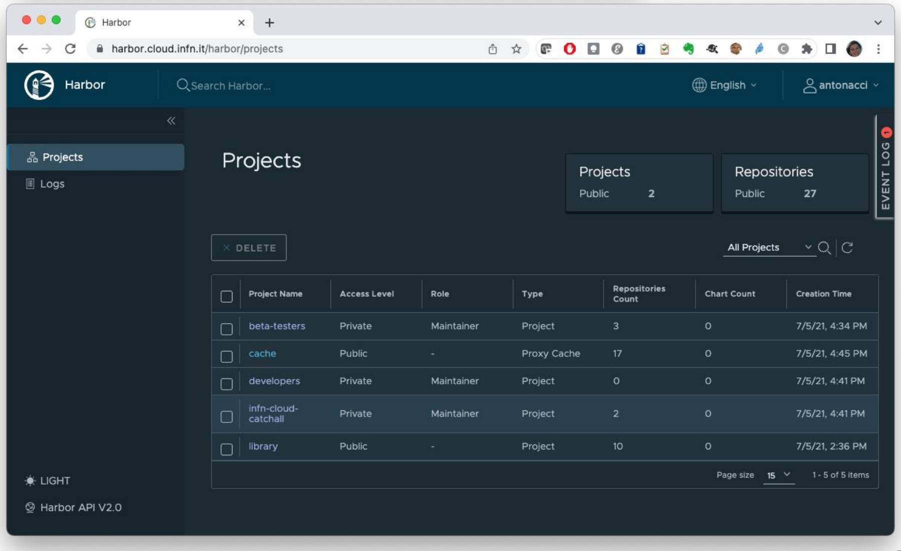
Self-Managed Services

Docker related services

How to manage and deploy containers on INFN Cloud

Harbor: docker registry





Project Name	Access Level	Role	Type	Repositories Count	Chart Count	Creation Time
beta-testers	Private	Maintainer	Project	3	0	7/5/21, 4:34 PM
cache	Public	-	Proxy Cache	17	0	7/5/21, 4:45 PM
developers	Private	Maintainer	Project	0	0	7/5/21, 4:41 PM
infn-cloud-catchall	Private	Maintainer	Project	2	0	7/5/21, 4:41 PM
library	Public	-	Project	10	0	7/5/21, 2:36 PM

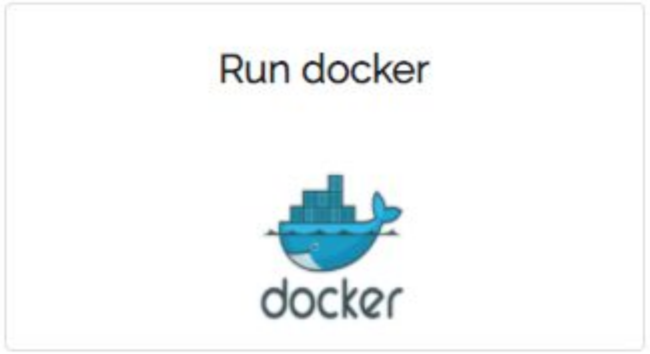
Page size 15 1 - 5 of 5 items

Two types of projects supported:

- **Public**: any user can pull images from this project (this is a convenient way to share repositories);
- **Private**: only users who are members of the project can pull images.

Proxy cache configured: when a pull request comes to a proxy cache project, if the image is not cached, Harbor pulls the image from the target registry and serves the pull command as if it is a local image from the proxy cache project.

https://harbor.cloud.infn.it

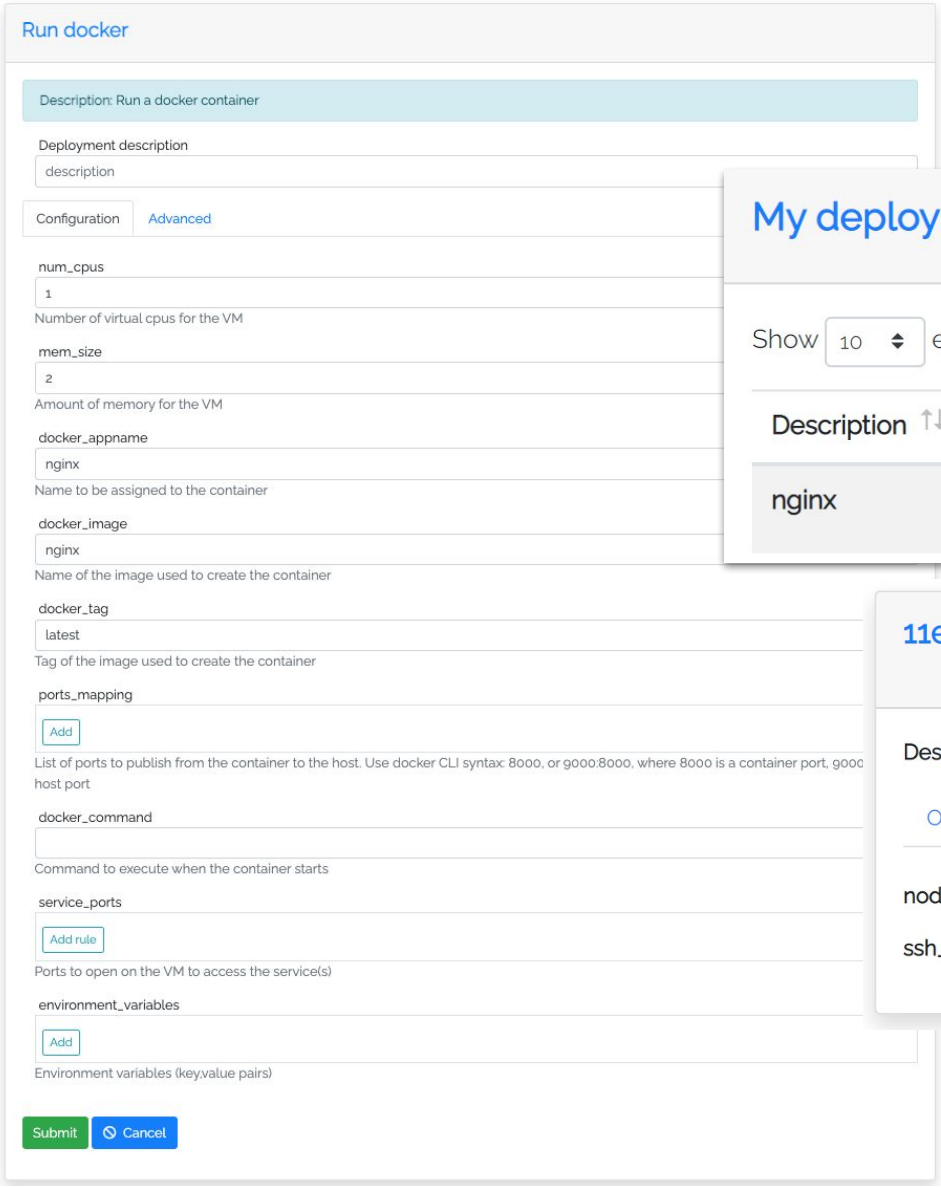


Docker run use-case

How to run a container on INFN Cloud

Configure your dockerized service

The configuration form allows you to customize your deployment.



Run docker

Description: Run a docker container

Deployment description	description
num_cpus	1
mem_size	2
docker_appname	nginx
docker_image	nginx
docker_tag	latest
ports_mapping	
docker_command	
service_ports	
environment_variables	

Number of virtual cpus for the VM

Amount of memory for the VM

Name to be assigned to the container

Name of the image used to create the container

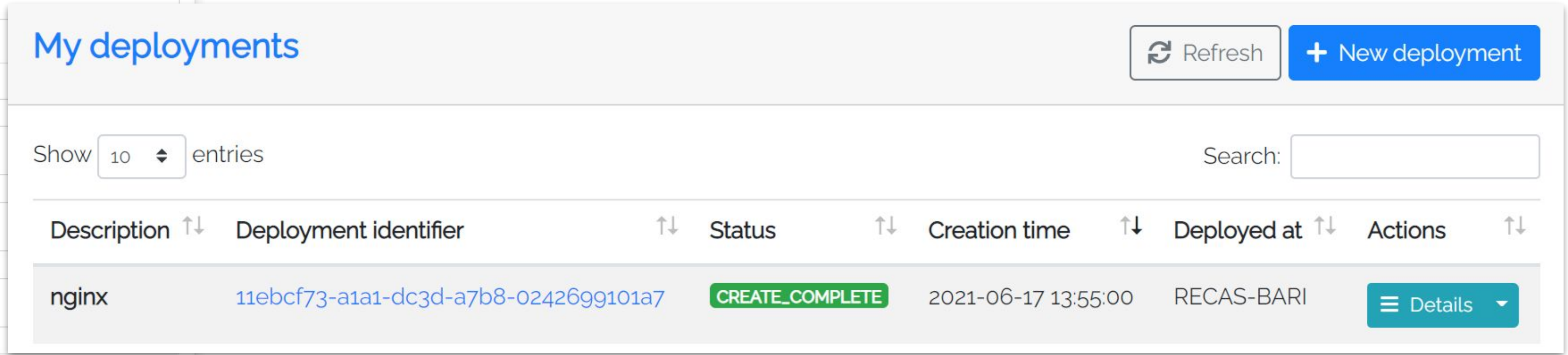
Tag of the image used to create the container

List of ports to publish from the container to the host. Use docker CLI syntax: 8000, or 9000:8000, where 8000 is a container port, 9000 host port

Command to execute when the container starts

Ports to open on the VM to access the service(s)

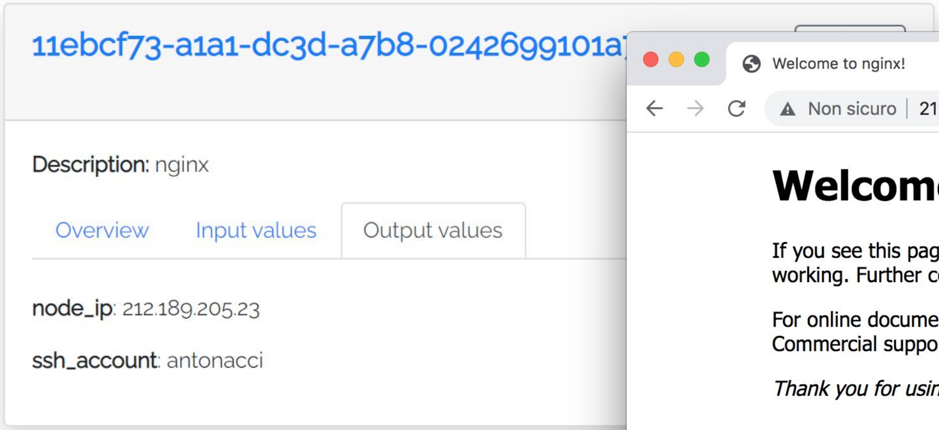
Environment variables (key,value pairs)



My deployments

Show 10 entries

Description	Deployment identifier	Status	Creation time	Deployed at	Actions
nginx	11ebcf73-a1a1-dc3d-a7b8-0242699101a7	CREATE_COMPLETE	2021-06-17 13:55:00	RECAS-BARI	Details

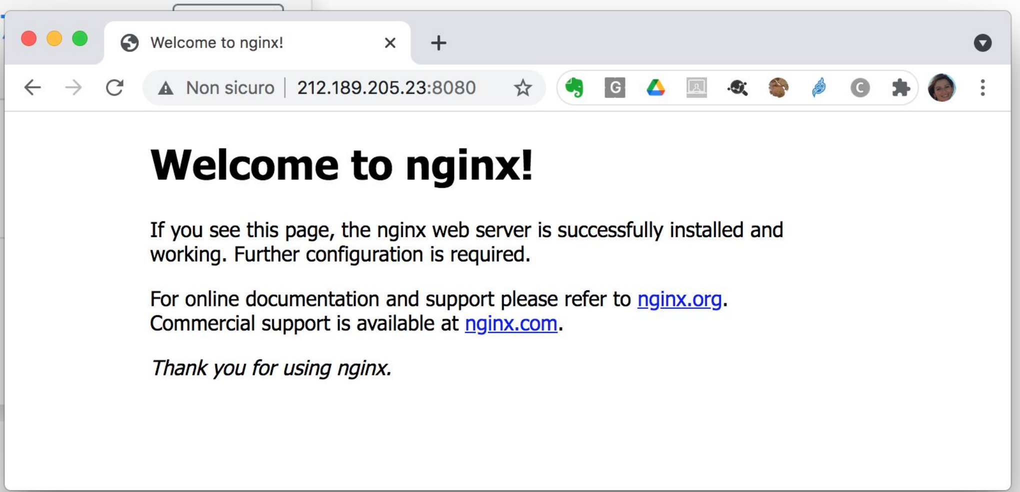


11ebcf73-a1a1-dc3d-a7b8-0242699101a7

Description: nginx

node_ip: 212.189.205.23

ssh_account: antonacci



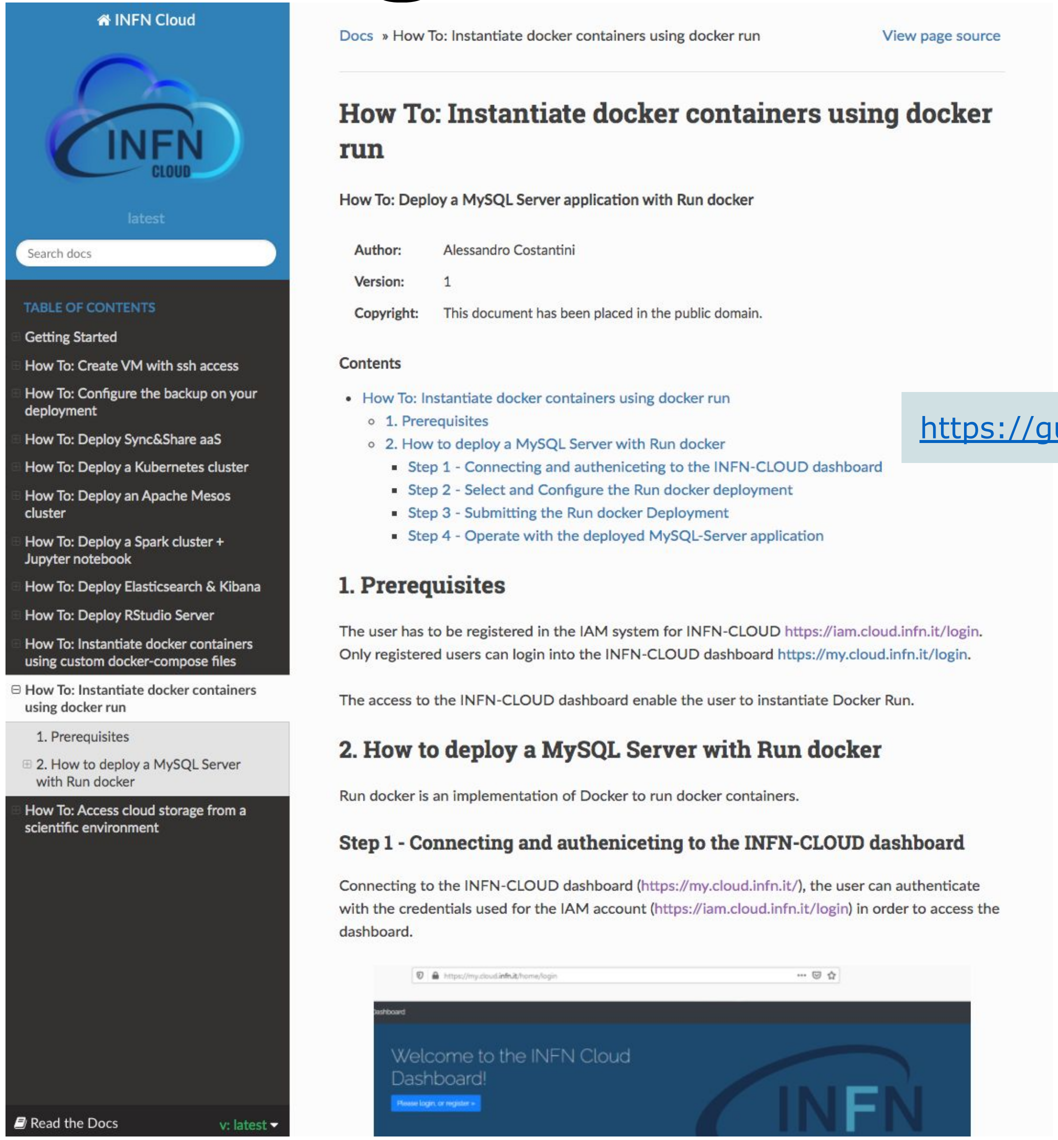
Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

How to su guides.cloud.infn.it



INFN Cloud

latest

Search docs

TABLE OF CONTENTS

Read the Docs v: latest

Docs » How To: Instantiate docker containers using docker run View page source

How To: Instantiate docker containers using docker run

How To: Deploy a MySQL Server application with Run docker

Author:	Alessandro Costantini
Version:	1
Copyright:	This document has been placed in the public domain.

Contents

1. Prerequisites

The user has to be registered in the IAM system for INFN-CLOUD https://iam.cloud.infn.it/login. Only registered users can login into the INFN-CLOUD dashboard https://my.cloud.infn.it/login.

The access to the INFN-CLOUD dashboard enable the user to instantiate Docker Run.

2. How to deploy a MySQL Server with Run docker

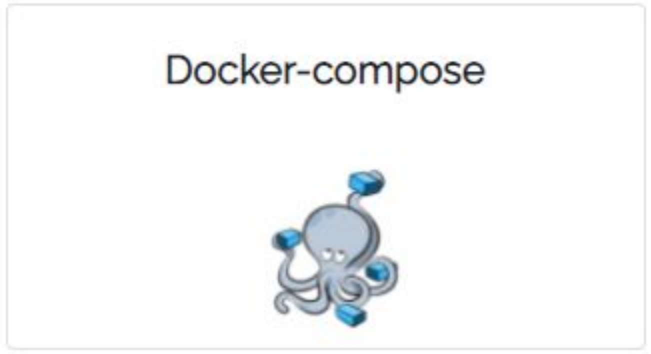
Run docker is an implementation of Docker to run docker containers.

Step 1 - Connecting and authenicenting to the INFN-CLOUD dashboard

Connecting to the INFN-CLOUD dashboard (https://my.cloud.infn.it/), the user can authenticate with the credentials used for the IAM account (https://iam.cloud.infn.it/login) in order to access the dashboard.

Welcome to the INFN Cloud Dashboard!

https://guides.cloud.infn.it/docs/users-guides/en/latest/users_guides/howto8.html

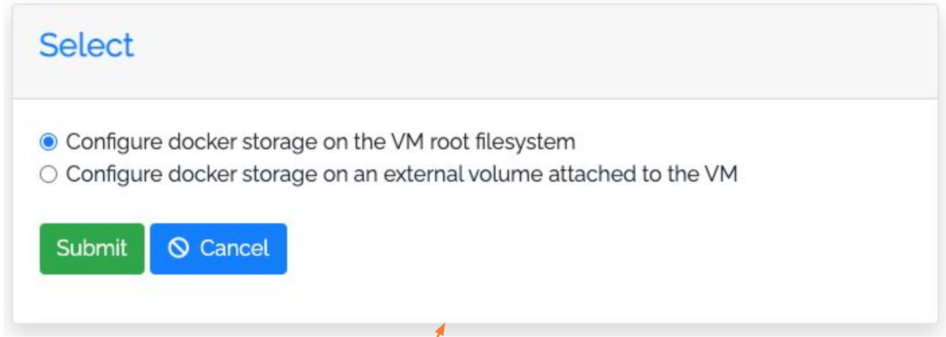


Docker-compose use-case

How to deploy a machine with docker compose pre-installed

and eventually run a docker-compose file fetched from a given URL

Configure your service

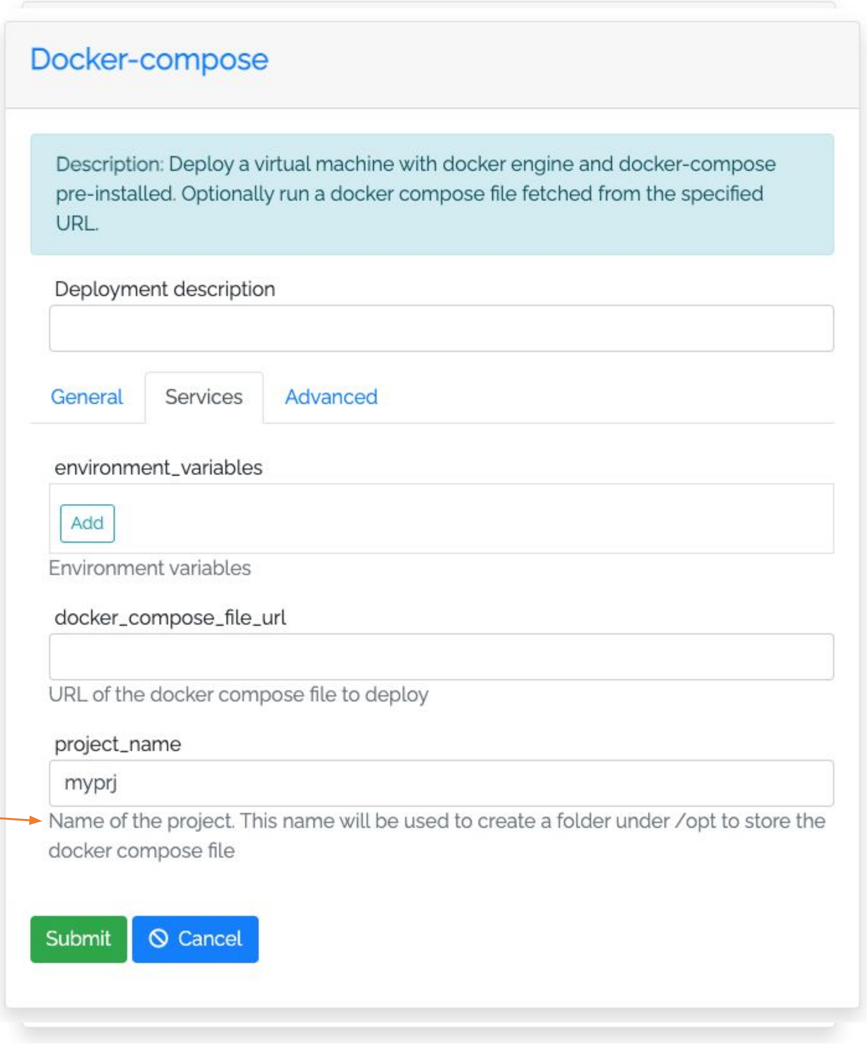


Select

- [✓ ●] Configure docker storage on the VM root filesystem
- ○ Configure docker storage on an external volume attached to the VM

You can choose to

- Put the docker storage on a separate volume
- Configure the machine with only docker and docker-compose or provide a docker compose file URL to start your services



Docker-compose

Description: Deploy a virtual machine with docker engine and docker-compose pre-installed. Optionally run a docker compose file fetched from the specified URL.

Deployment description

environment_variables

Environment variables

docker_compose_file_url

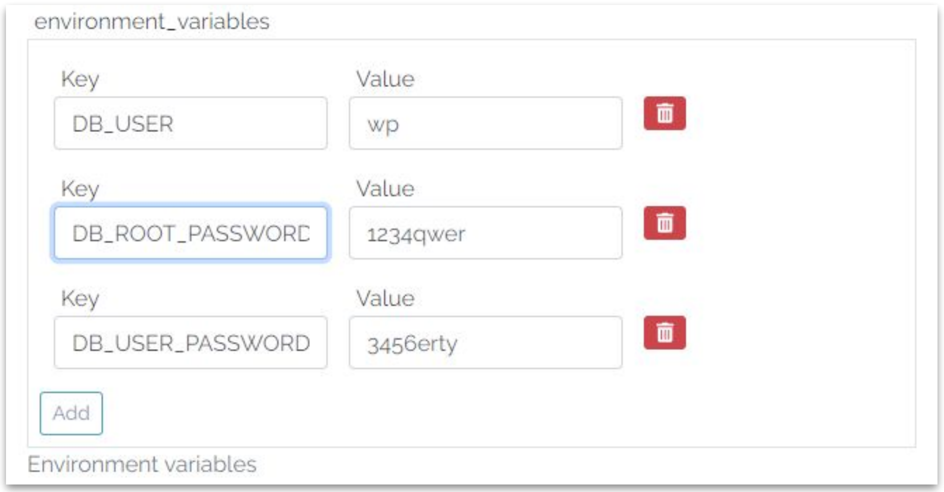
URL of the docker compose file to deploy

project_name

myprj

Name of the project. This name will be used to create a folder under /opt to store the docker compose file

Environment variables management



environment_variables

Key	Value
DB_USER	wp
DB_ROOT_PASSWORD	1234qwer
DB_USER_PASSWORD	3456erty

Add

Environment variables

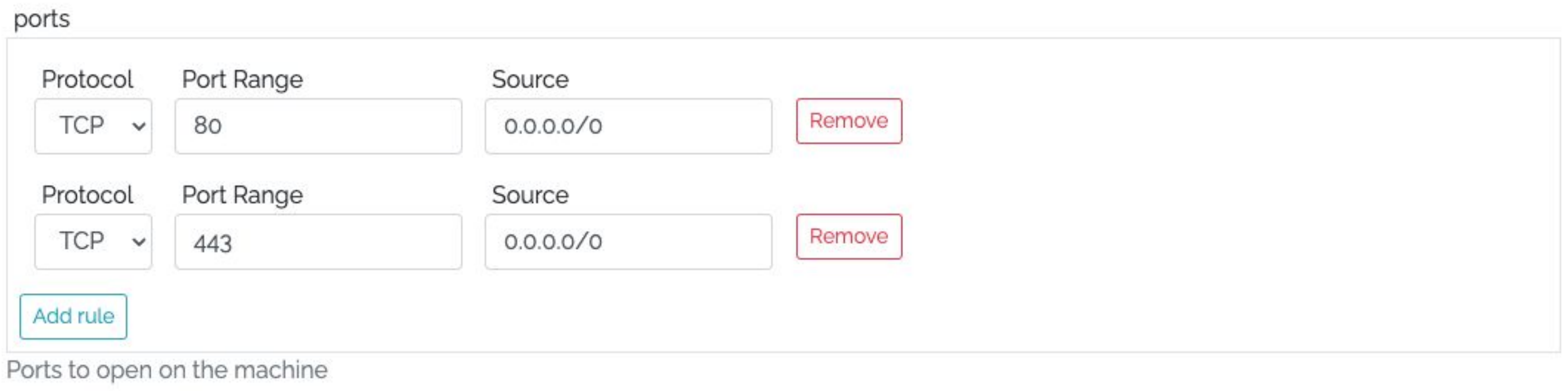
```
services:
  ........
  app:
    depends_on:
      - db
    image: wordpress
    container_name: app
    volumes:
      - wp-content:/var/www/html/wp-content
    environment:
      - WORDPRESS_DB_HOST=db:3306
      - WORDPRESS_DB_USER=${DB_USER}
      - WORDPRESS_DB_PASSWORD=${DB_USER_PASSWORD}
      - VIRTUAL_HOST=wp.${HOST_PUBLIC_IP}.myip.cloud.infn.it
    expose:
      - 80
```

- The special variable *HOST_PUBLIC_IP* is made available by the PaaS system and contains the public IP assigned to the VM

- This env variable can be used as a normal env variable inside the user docker compose file

Ports management

You can define the set of ports that must be automatically opened on the server in order to access your services


ports

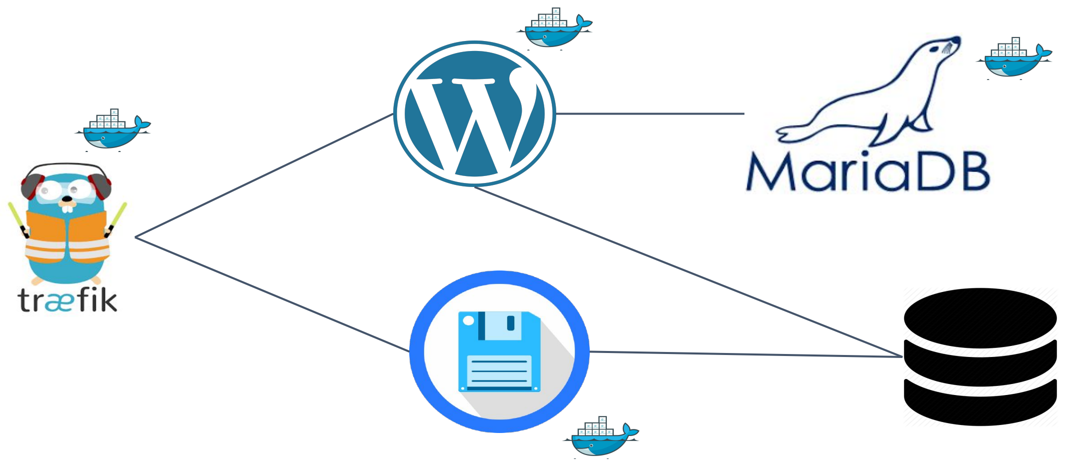
Protocol	Port Range	Source	
TCP	80	0.0.0.0/0	Remove
TCP	443	0.0.0.0/0	Remove

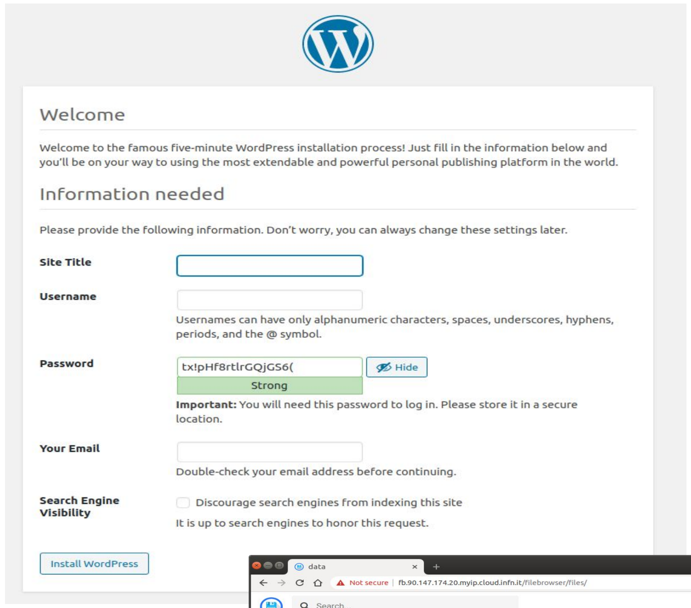
Ports to open on the machine

Docker compose example

https://baltig.infn.it/infn-cloud/apps/-/blob/master/compose-example/docker-demo.yaml

Author: Stefano Stalio (LNGS)





Welcome

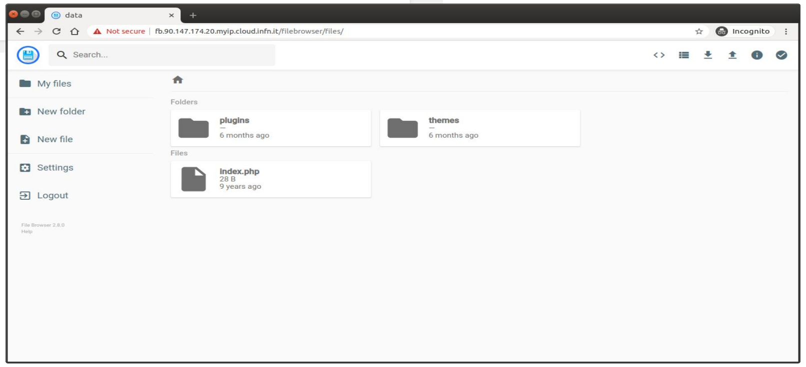
Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

Information needed

Please provide the following information. Don't worry, you can always change these settings later.

Site Title	
Username	Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.
Password	tx!pHf8rtlrGQjGS6(Hide Strong **Important:** You will need this password to log in. Please store it in a secure location.
Your Email	Double-check your email address before continuing.
Search Engine Visibility	Discourage search engines from indexing this site It is up to search engines to honor this request.

Install WordPress



data

Not secure | fb.90.147.174.20.myip.cloud.infn.it/filebrowser/files/

Incognito

Search...

My files
New folder
New file
Settings
Logout

Folders

plugins — 6 months ago

themes — 6 months ago

Files

index.php 28 B 9 years ago

File Browser 2.8.0
Help

DNS @INFN Cloud

INFN Cloud provides a DNSaaS mechanism that associates a DNS name to each VM public IP

```
$ host wp.90.147.174.132.myip.cloud.infn.it
wp.90.147.174.132.myip.cloud.infn.it has address
90.147.174.132
```

This mechanism is based on xip.io (wildcard DNS) and is exploited for the automatic generation of ssl certificates (e.g. with letsencrypt)

```
services:
 db:
  image: mariadb
  container_name: db
  volumes:
   - db:/var/lib/mysql
  environment:
   - MYSQL_ROOT_PASSWORD=${DB_ROOT_PASSWORD}
   - MYSQL_DATABASE=wordpress
   - MYSQL_USER=${DB_USER}
   - MYSQL_PASSWORD=${DB_USER_PASSWORD}
  expose:
   - 3306
 app:
  depends_on:
   - db
  image: wordpress
  container_name: app
  volumes:
   - wp-content:/var/www/html/wp-content
  environment:
   - WORDPRESS_DB_HOST=db:3306
   - WORDPRESS_DB_USER=${DB_USER}
   - WORDPRESS_DB_PASSWORD=${DB_USER_PASSWORD}
   - VIRTUAL_HOST=wp.${HOST_PUBLIC_IP}.myip.cloud.infn.it
  expose:
   - 80
```

SSL Terminator & Load-balancer

- You can use Traefik as load balancer and SSL terminator. https://traefik.io/traefik/

- Traefik is able to renew letsencrypt certificates

```
services:
 load_balancer:
  image: traefik
  container_name: traefik
  volumes:
   - letsencrypt:/letsencrypt
   - /var/run/docker.sock:/var/run/docker.sock:ro
  ports:
   - "80:80"
   - "443:443"
  command:
   - "--api.insecure=true"
   - "--providers.docker=true"
   - "--providers.docker.exposedbydefault=false"
   - "--entrypoints.web.address=:80"
   - "--entrypoints.websecure.address=:443"
   - "--certificatesresolvers.myhttpchallenge.acme.httpchallenge=true"
   -
"--certificatesresolvers.myhttpchallenge.acme.httpchallenge.entrypoint=w
eb"
   -
"--certificatesresolvers.myhttpchallenge.acme.email=${CONTACT_EMAIL}"
   -
"--certificatesresolvers.myhttpchallenge.acme.storage=/letsencrypt/acme.
json"
```

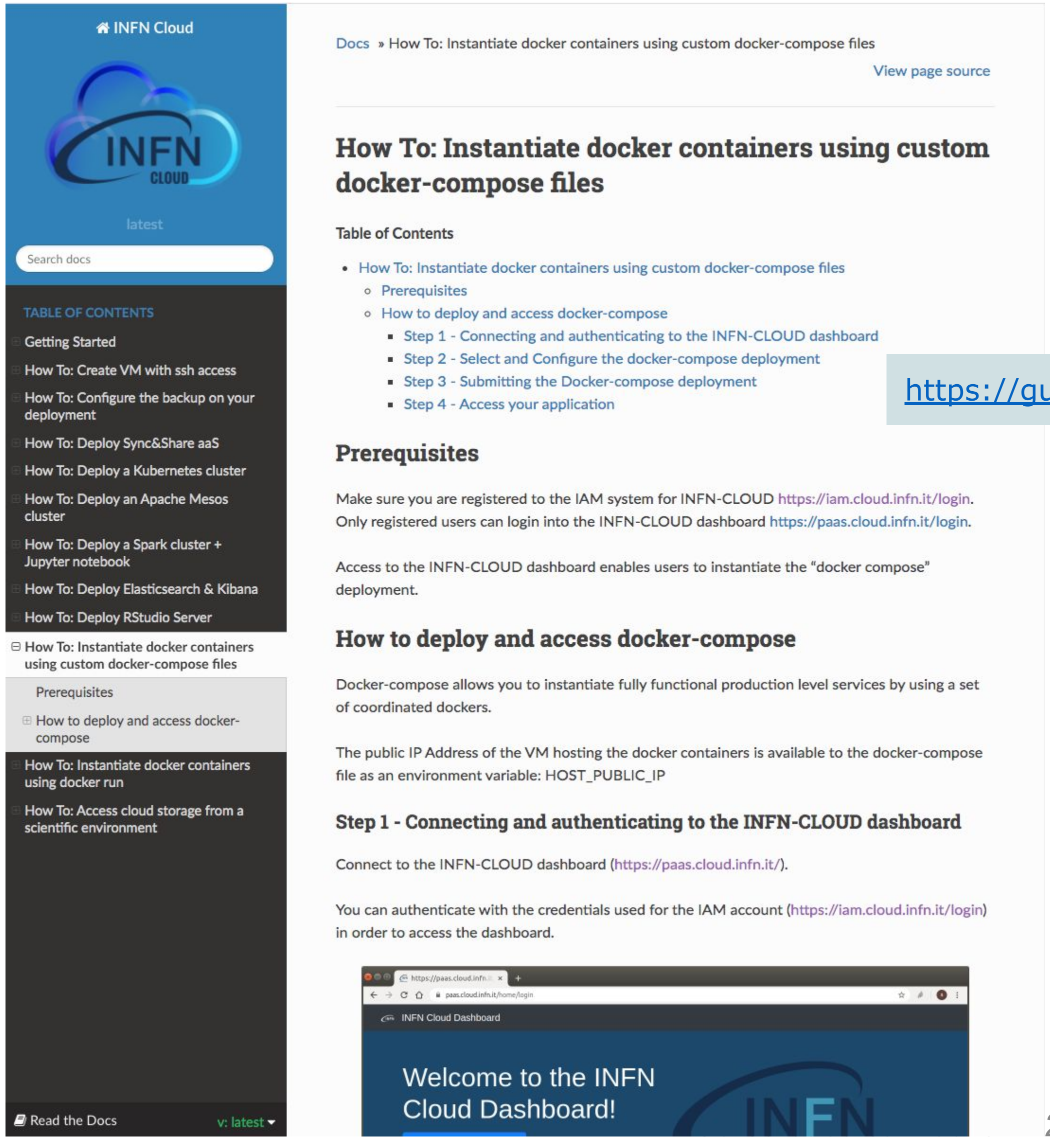
Traefik configuration

Traefik is automatically configured through the labels* exposed by the containers

(*) “A label is a **key=value** *pair that applies metadata to a container.”*

```
services:
  app:
   depends_on:
     - db
   image: wordpress
   container_name: app
   volumes:
     - wp-content:/var/www/html/wp-content
   environment:
     - WORDPRESS_DB_HOST=db:3306
     - WORDPRESS_DB_USER=${DB_USER}
     - WORDPRESS_DB_PASSWORD=${DB_USER_PASSWORD}
     - VIRTUAL_HOST=wp.${HOST_PUBLIC_IP}.myip.cloud.infn.it
   expose:
     - 80
   labels:
     - "traefik.enable=true"
     - "traefik.http.middlewares.app-redirect-ssl.redirectscheme.scheme=https"
     - "traefik.http.routers.app-nossl.middlewares=app-redirect-ssl"
     -
"traefik.http.routers.app-nossl.rule=Host(`wp.${HOST_PUBLIC_IP}.myip.cloud.infn.
it`)"
     - "traefik.http.routers.app-nossl.entrypoints=web"
     -
"traefik.http.routers.app.rule=Host(`wp.${HOST_PUBLIC_IP}.myip.cloud.infn.it`)"
     - "traefik.http.routers.app.entrypoints=websecure"
     - "traefik.http.routers.app.tls.certresolver=myhttpchallenge"
     - "traefik.http.routers.app.tls=true"
```

How to su guides.cloud.infn.it



INFN Cloud

latest

Search docs

TABLE OF CONTENTS

- Getting Started
- How To: Create VM with ssh access
- How To: Configure the backup on your deployment
- How To: Deploy Sync&Share aaS
- How To: Deploy a Kubernetes cluster
- How To: Deploy an Apache Mesos cluster
- How To: Deploy a Spark cluster + Jupyter notebook
- How To: Deploy Elasticsearch & Kibana
- How To: Deploy RStudio Server
- How To: Instantiate docker containers using custom docker-compose files
  - Prerequisites
  - How to deploy and access docker-compose
- How To: Instantiate docker containers using docker run
- How To: Access cloud storage from a scientific environment

Read the Docs v: latest

Docs » How To: Instantiate docker containers using custom docker-compose files

View page source

How To: Instantiate docker containers using custom docker-compose files

Table of Contents

- How To: Instantiate docker containers using custom docker-compose files
  - Prerequisites
  - How to deploy and access docker-compose
    - Step 1 - Connecting and authenticating to the INFN-CLOUD dashboard
    - Step 2 - Select and Configure the docker-compose deployment
    - Step 3 - Submitting the Docker-compose deployment
    - Step 4 - Access your application

Prerequisites

Make sure you are registered to the IAM system for INFN-CLOUD https://iam.cloud.infn.it/login. Only registered users can login into the INFN-CLOUD dashboard https://paas.cloud.infn.it/login.

Access to the INFN-CLOUD dashboard enables users to instantiate the "docker compose" deployment.

How to deploy and access docker-compose

Docker-compose allows you to instantiate fully functional production level services by using a set of coordinated dockers.

The public IP Address of the VM hosting the docker containers is available to the docker-compose file as an environment variable: HOST_PUBLIC_IP

Step 1 - Connecting and authenticating to the INFN-CLOUD dashboard

Connect to the INFN-CLOUD dashboard (https://paas.cloud.infn.it/).

You can authenticate with the credentials used for the IAM account (https://iam.cloud.infn.it/login) in order to access the dashboard.

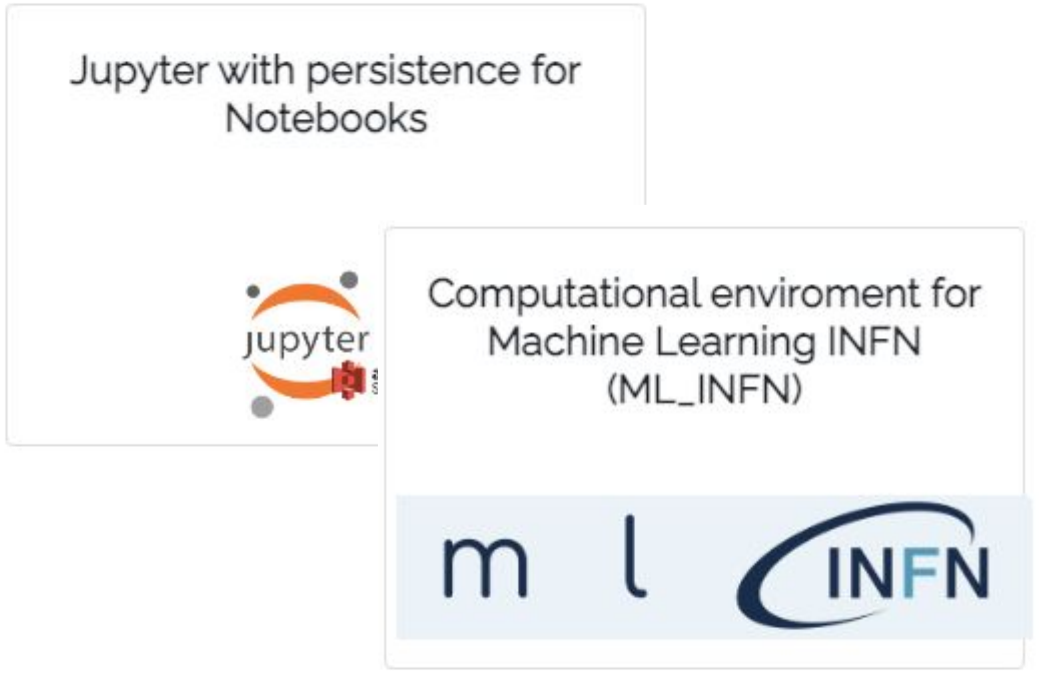
INFN Cloud Dashboard

Welcome to the INFN Cloud Dashboard!

https://guides.cloud.infn.it/docs/users-guides/en/latest/users_guides/howto7.html

Docker-based Advanced use-cases:

Multi-users JupyterHub
- **With Persistent storage**
- **With access to GPUs**
- **....**

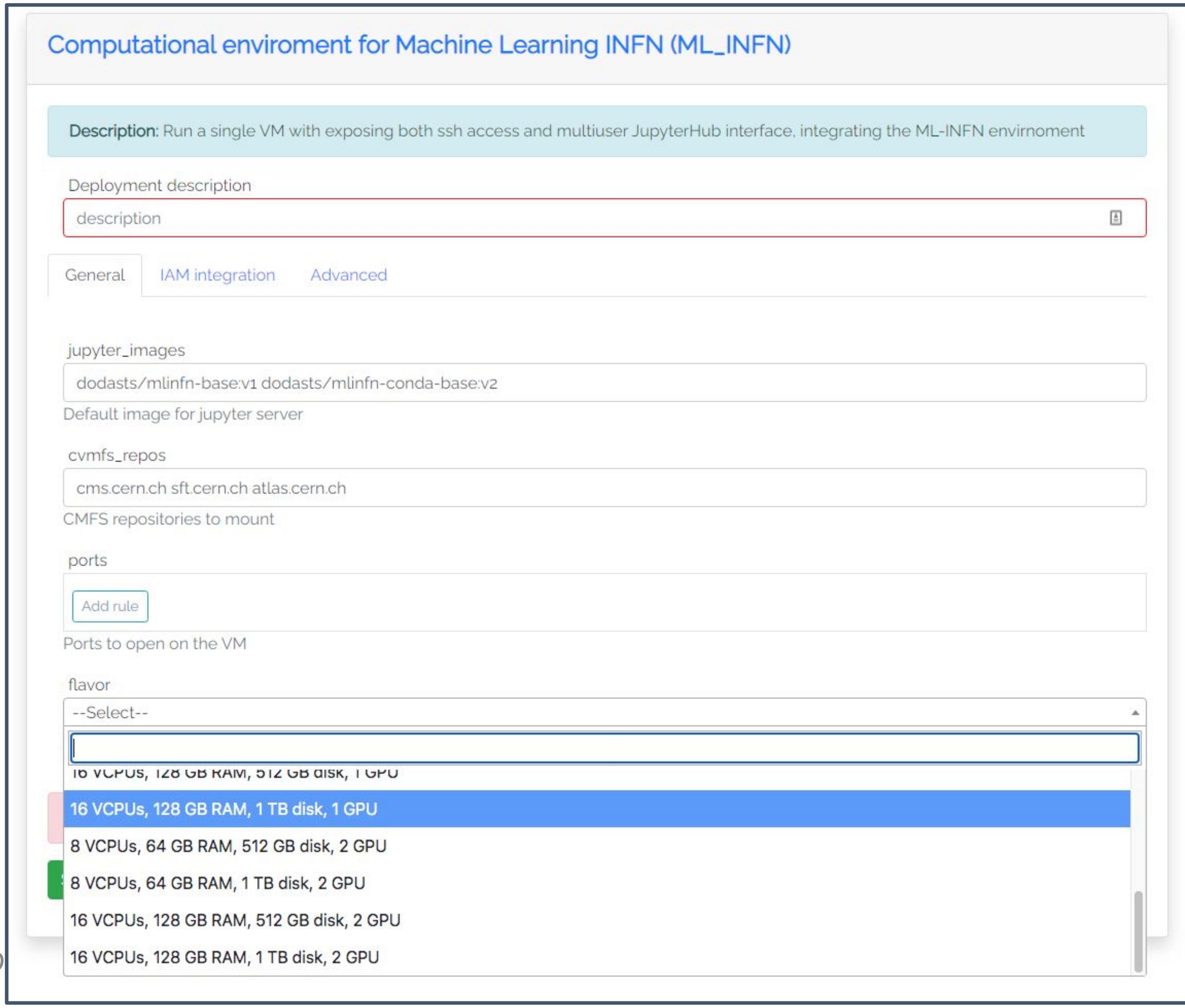

Jupyter with persistence for Notebooks

Computational enviroment for Machine Learning INFN (ML_INFN)

If you are authorized ... you can create your own machine!

Simple high-level configuration template to create your personal environment

- Either for single user and multi users (group activities)
  - Authorization based on IAM groups
- Ask for CVMFS areas, GPUs, ...


Computational enviroment for Machine Learning INFN (ML_INFN)

Description: Run a single VM with exposing both ssh access and multiuser JupyterHub interface, integrating the ML-INFN envirnoment

Deployment description

description

jupyter_images

dodasts/mlinfn-base:v1 dodasts/mlinfn-conda-base:v2

Default image for jupyter server

cvmfs_repos

cms.cern.ch sft.cern.ch atlas.cern.ch

CMFS repositories to mount

ports

Ports to open on the VM

flavor

--Select--

16 VCPUs, 128 GB RAM, 512 GB disk, 1 GPU

16 VCPUs, 128 GB RAM, 1 TB disk, 1 GPU

8 VCPUs, 64 GB RAM, 512 GB disk, 2 GPU

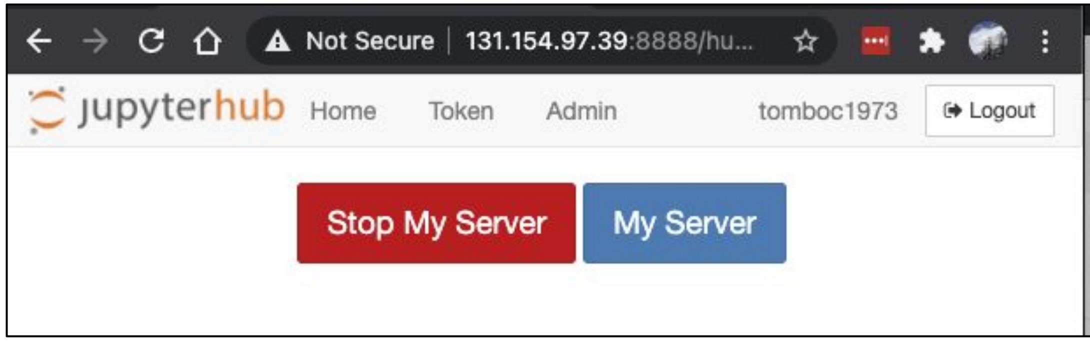
8 VCPUs, 64 GB RAM, 1 TB disk, 2 GPU

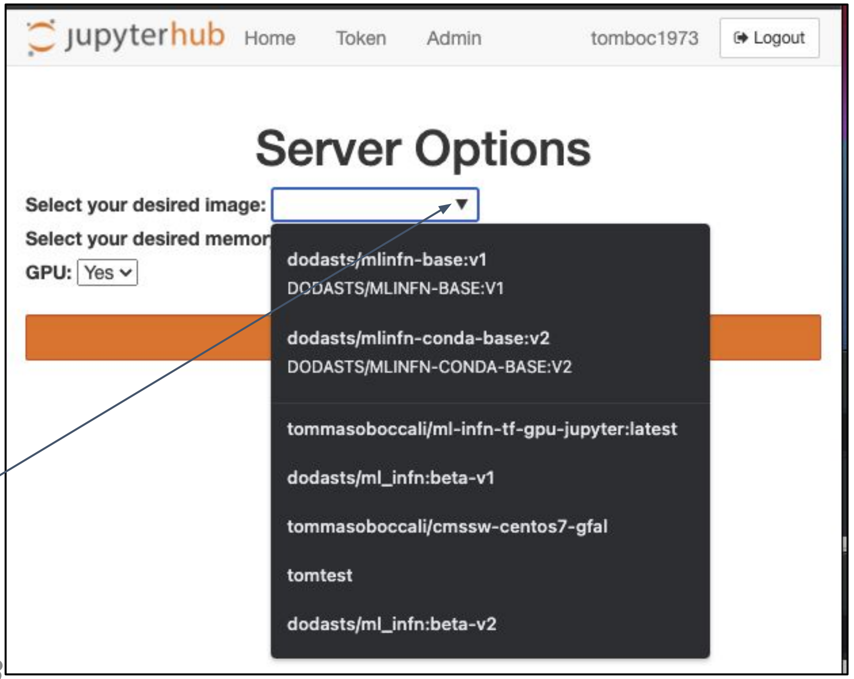
16 VCPUs, 128 GB RAM, 512 GB disk, 2 GPU

16 VCPUs, 128 GB RAM, 1 TB disk, 2 GPU

What is inside the VM?

- A **jupyterhub** runs in the VM, and allows **authorized users** to create their running instance through a container (taken either locally, or directly from dockerhub)
- All these containers use the resources of the VM, which are then shared for the user group
- Containers are accessible both via Jupyter Notebooks and via terminal (for the moment via browser)
- The administrator (owner of the service) can access the VM both ssh and via browser





Server Options

Select your desired image:

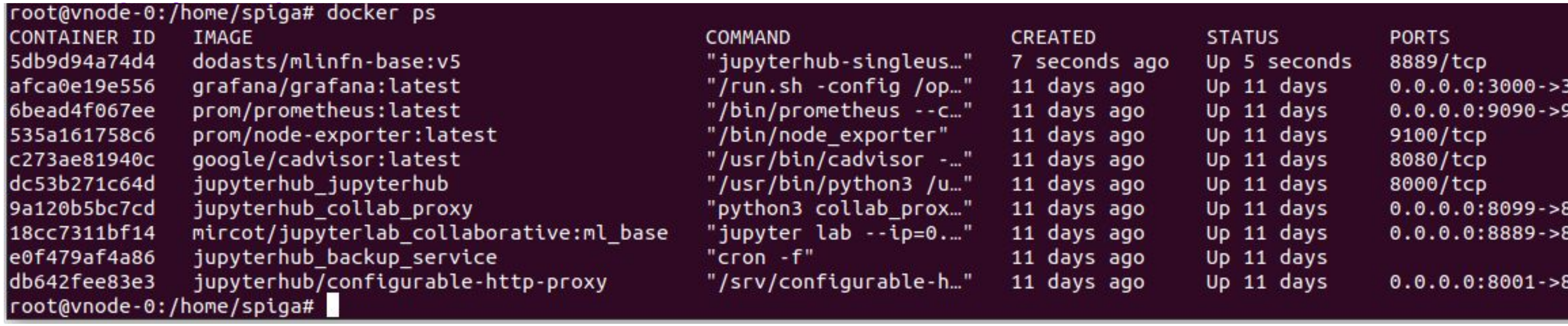
Select your desired memor

GPU: Yes

dodasts/mlinfn-base:v1
DODASTS/MLINFN-BASE:V1
dodasts/mlinfn-conda-base:v2
DODASTS/MLINFN-CONDA-BASE:V2
tommasoboccali/ml-infn-tf-gpu-jupyter:latest
dodasts/ml_infn:beta-v1
tommasoboccali/cmssw-centos7-gfal
tomtest
dodasts/ml_infn:beta-v2

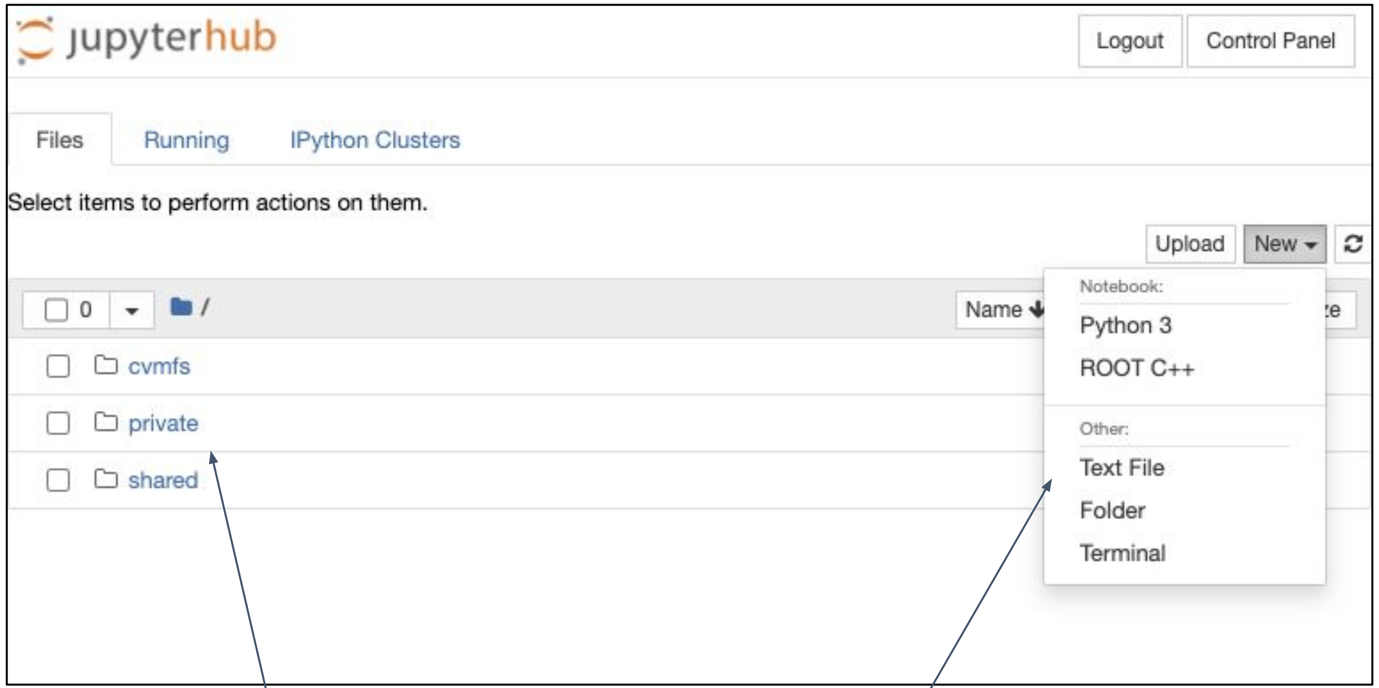
Here you can specify your image

How it is made:



```
root@vnode-0:/home/spiga# docker ps
CONTAINER ID   IMAGE                                         COMMAND                  CREATED          STATUS          PORTS
5db9d94a74d4   dodasts/mlinfn-base:v5                        "jupyterhub-singleus…"   7 seconds ago    Up 5 seconds    8889/tcp
afca0e19e556   grafana/grafana:latest                        "/run.sh -config /op…"   11 days ago      Up 11 days      0.0.0.0:3000->3
6bead4f067ee   prom/prometheus:latest                        "/bin/prometheus --c…"   11 days ago      Up 11 days      0.0.0.0:9090->9
535a161758c6   prom/node-exporter:latest                     "/bin/node_exporter"     11 days ago      Up 11 days      9100/tcp
c273ae81940c   google/cadvisor:latest                        "/usr/bin/cadvisor -…"   11 days ago      Up 11 days      8080/tcp
dc53b271c64d   jupyterhub_jupyterhub                         "/usr/bin/python3 /u…"   11 days ago      Up 11 days      8000/tcp
9a120b5bc7cd   jupyterhub_collab_proxy                       "python3 collab_prox…"   11 days ago      Up 11 days      0.0.0.0:8099->8
18cc7311bf14   mircot/jupyterlab_collaborative:ml_base       "jupyter lab --ip=0.…"   11 days ago      Up 11 days      0.0.0.0:8889->8
e0f479af4a86   jupyterhub_backup_service                     "cron -f"                11 days ago      Up 11 days
db642fee83e3   jupyterhub/configurable-http-proxy            "/srv/configurable-h…"   11 days ago      Up 11 days      0.0.0.0:8001->8
root@vnode-0:/home/spiga#
```

Access as “User”

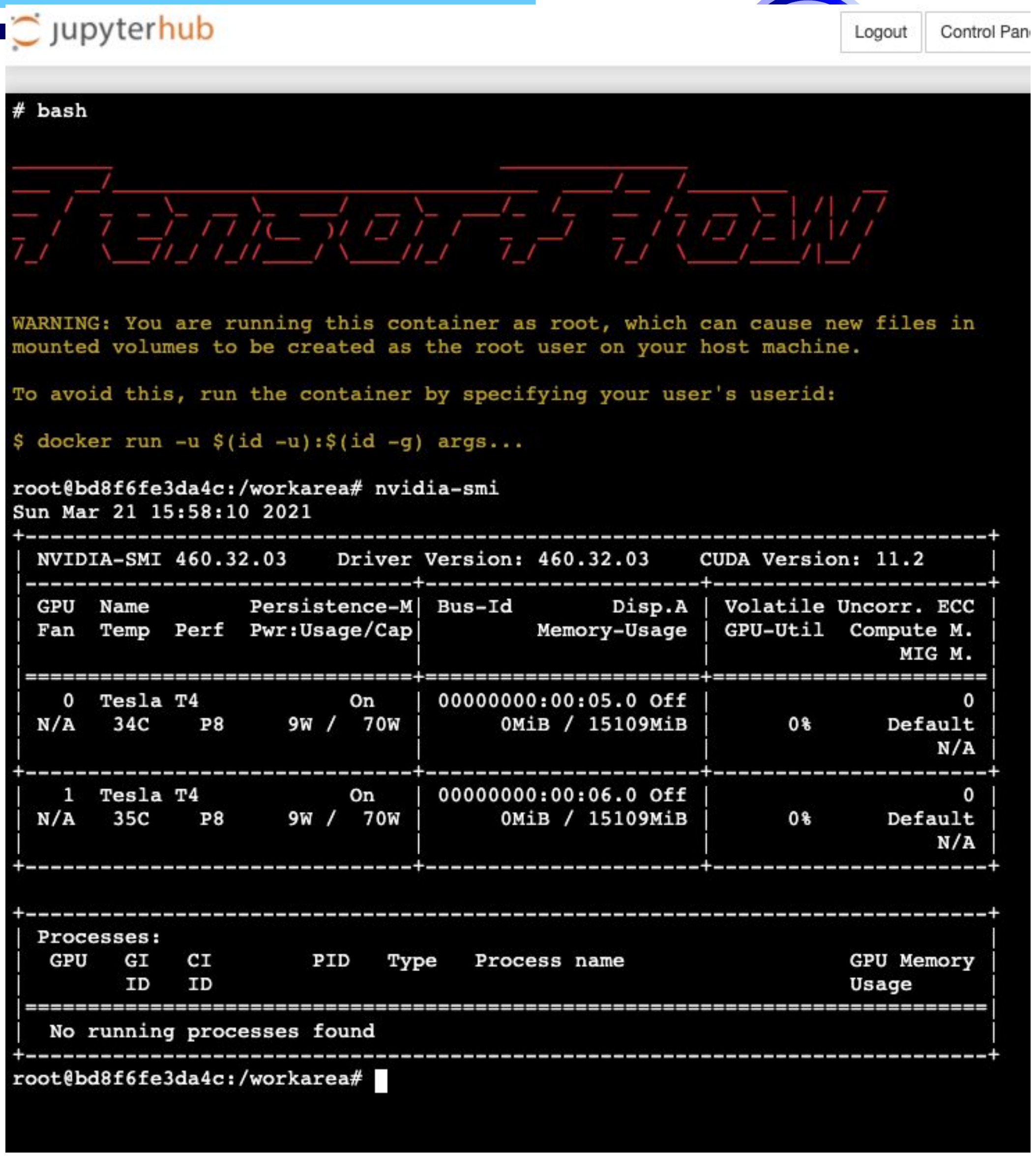


Select items to perform actions on them.

cvmfs

private

shared



```
# bash

WARNING: You are running this container as root, which can cause new files in
mounted volumes to be created as the root user on your host machine.

To avoid this, run the container by specifying your user's userid:

$ docker run -u $(id -u):$(id -g) args...

root@bd8f6fe3da4c:/workarea# nvidia-smi
Sun Mar 21 15:58:10 2021
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 460.32.03    Driver Version: 460.32.03    CUDA Version: 11.2     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  Tesla T4            On   | 00000000:00:05.0 Off |                    0 |
| N/A   34C    P8     9W /  70W |      0MiB / 15109MiB |      0%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+
|   1  Tesla T4            On   | 00000000:00:06.0 Off |                    0 |
| N/A   35C    P8     9W /  70W |      0MiB / 15109MiB |      0%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|  No running processes found                                                 |
+-----------------------------------------------------------------------------+
root@bd8f6fe3da4c:/workarea# 
```

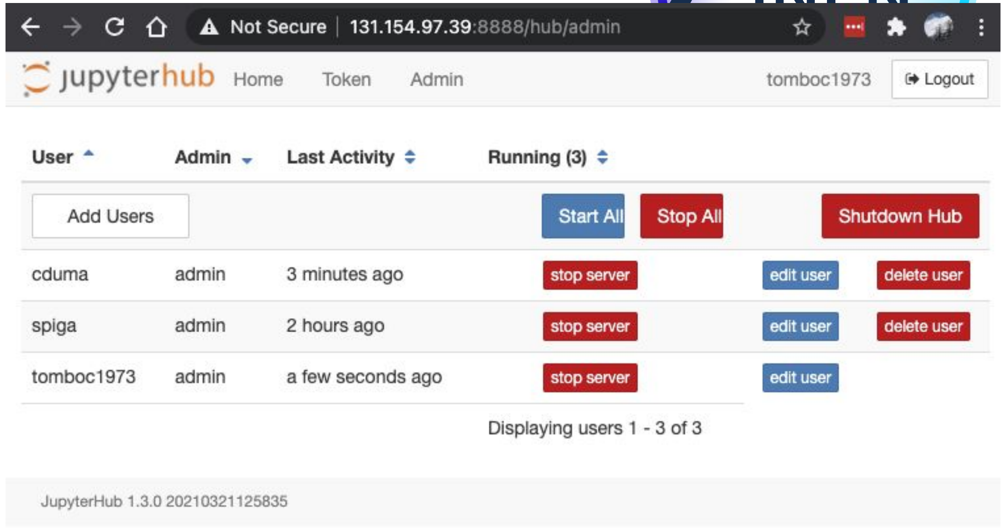
Areas “cvmfs” and “shared” are shared with all the users of the VM

Access granted via notebooks and via terminal

Root access, 2 GPUs available

Monitoring etc

- The administrator can manage containers
- All users can see detailed monitoring information



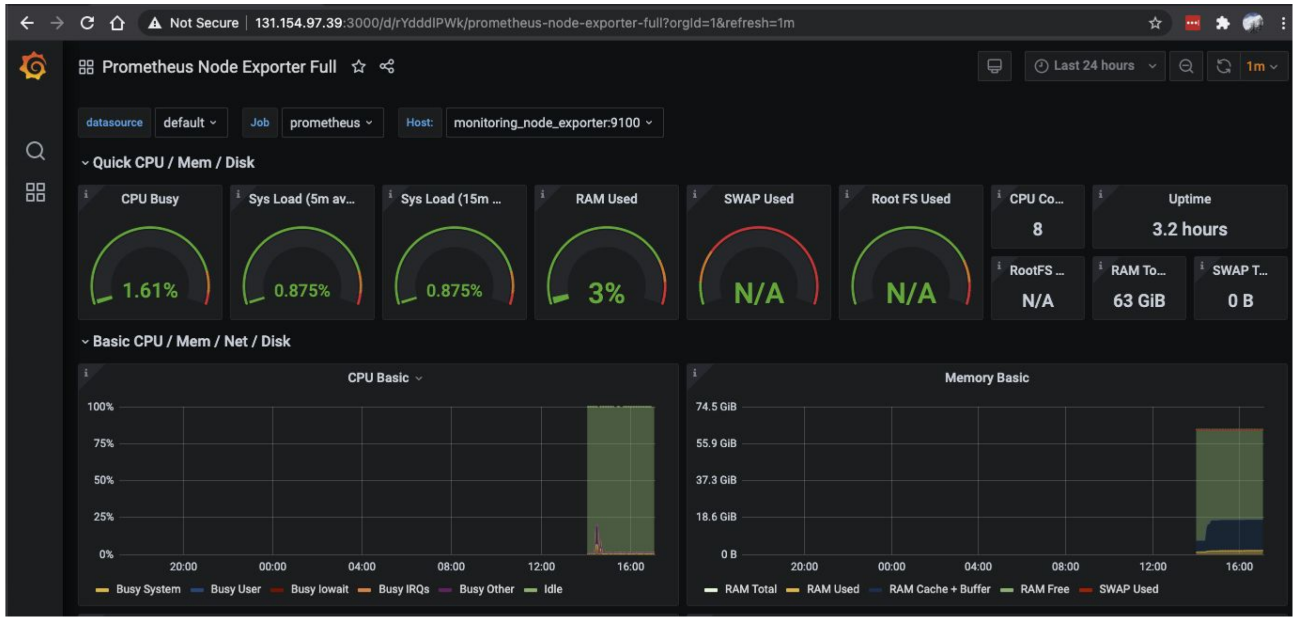
Not Secure | 131.154.97.39:8888/hub/admin

jupyterhub Home Token Admin tomboc1973 Logout

User	Admin	Last Activity	Running (3)		
Add Users			Start All Stop All		Shutdown Hub
cduma	admin	3 minutes ago	stop server	edit user	delete user
spiga	admin	2 hours ago	stop server	edit user	delete user
tomboc1973	admin	a few seconds ago	stop server	edit user	

Displaying users 1 - 3 of 3

JupyterHub 1.3.0 20210321125835

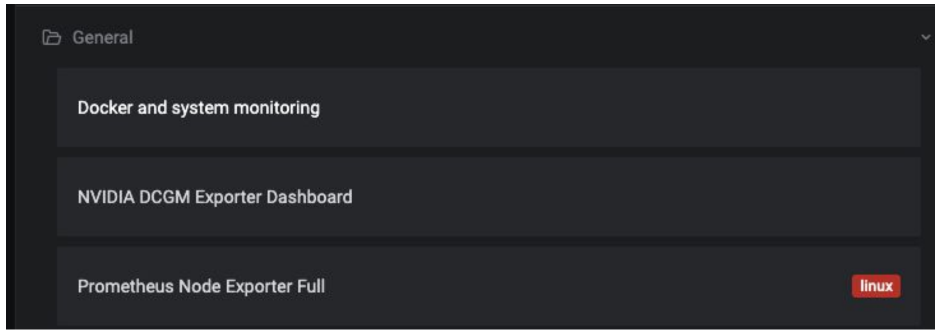


Not Secure | 131.154.97.39:3000/d/rYdddIPWk/prometheus-node-exporter-full?orgId=1&refresh=1m

Prometheus Node Exporter Full

Last 24 hours 1m

datasource: default · Job: prometheus · Host: monitoring_node_exporter:9100



General

Docker and system monitoring

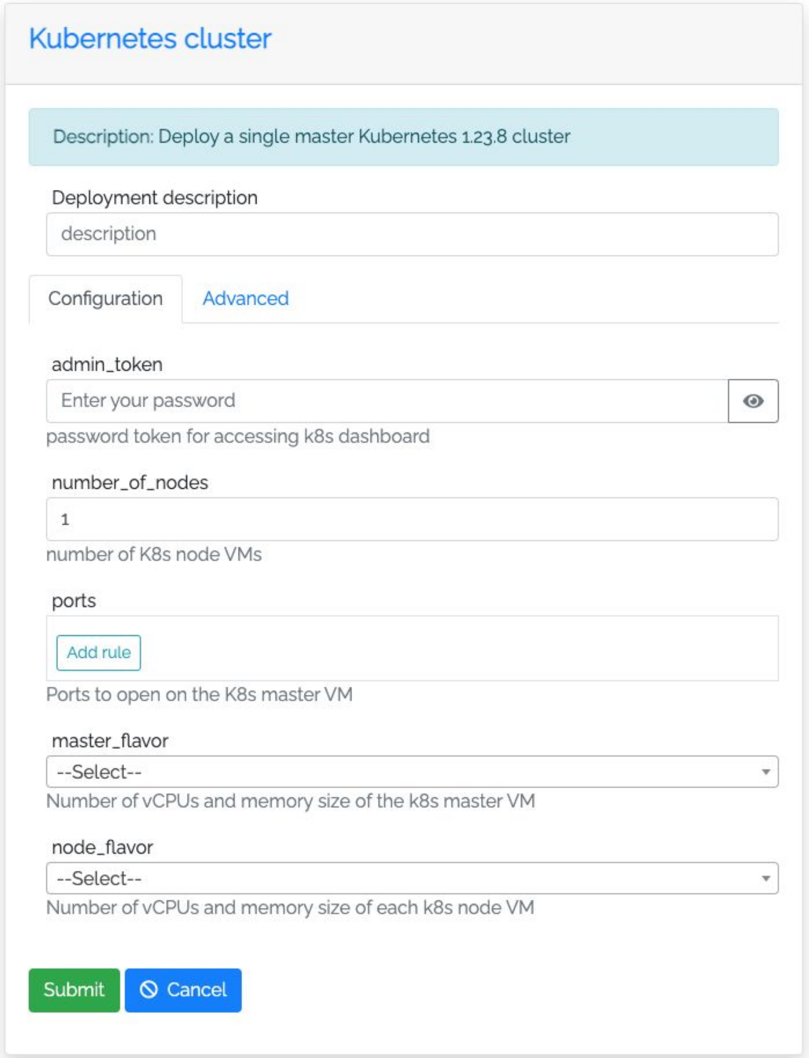
NVIDIA DCGM Exporter Dashboard

Prometheus Node Exporter Full linux

Kubernetes cluster use-case

How to deploy a complete k8s cluster on INFN Cloud

Configure your cluster


Kubernetes cluster

Description: Deploy a single master Kubernetes 1.23.8 cluster

Deployment description

description

admin_token

Enter your password

password token for accessing k8s dashboard

number_of_nodes

1

number of K8s node VMs

ports

Ports to open on the K8s master VM

master_flavor

--Select--

Number of vCPUs and memory size of the k8s master VM

node_flavor

--Select--

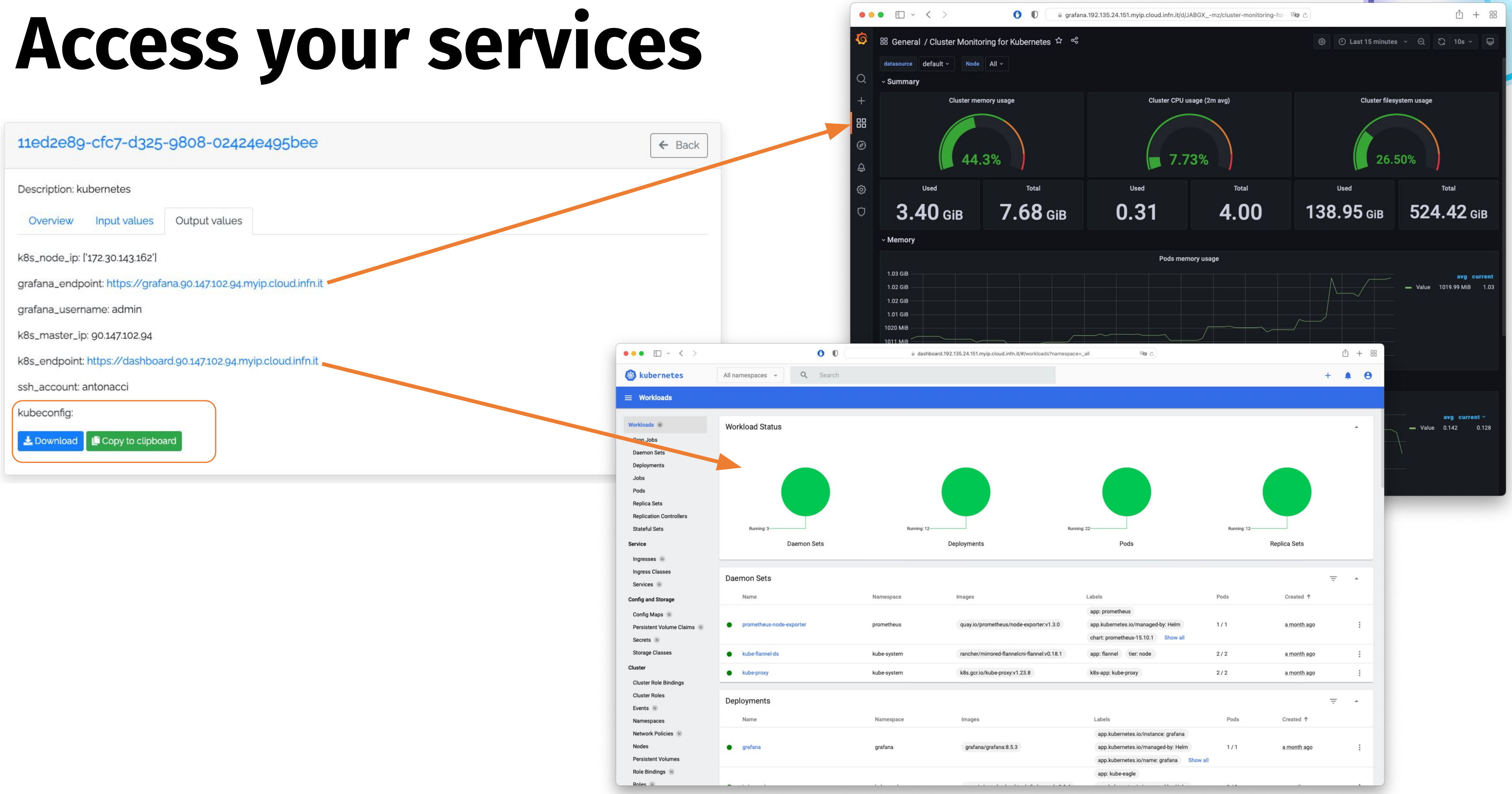
Number of vCPUs and memory size of each k8s node VM

The configuration form allows you to customize your cluster:

- Number of nodes
- Ports to be opened on the master node
- Flavor for the master and node servers

Nodes with GPUs can be spawned for specific projects (e.g. ML-INFN)

Access your services



11ed2e89-cfc7-d325-9808-02424e495bee

Description: kubernetes

k8s_node_ip: ['172.30.143.162']

grafana_endpoint: https://grafana.90.147.102.94.myip.cloud.infn.it

grafana_username: admin

k8s_master_ip: 90.147.102.94

k8s_endpoint: https://dashboard.90.147.102.94.myip.cloud.infn.it

ssh_account: antonacci

kubeconfig:

General / Cluster Monitoring for Kubernetes

Summary

Cluster memory usage		Cluster CPU usage (2m avg)		Cluster filesystem usage	
44.3%		7.73%		26.50%	
Used	Total	Used	Total	Used	Total
3.40 GiB	7.68 GiB	0.31	4.00	138.95 GiB	524.42 GiB

Memory

Pods memory usage

Workload Status

Daemon Sets	Deployments	Pods	Replica Sets
Running: 3	Running: 12	Running: 22	Running: 12

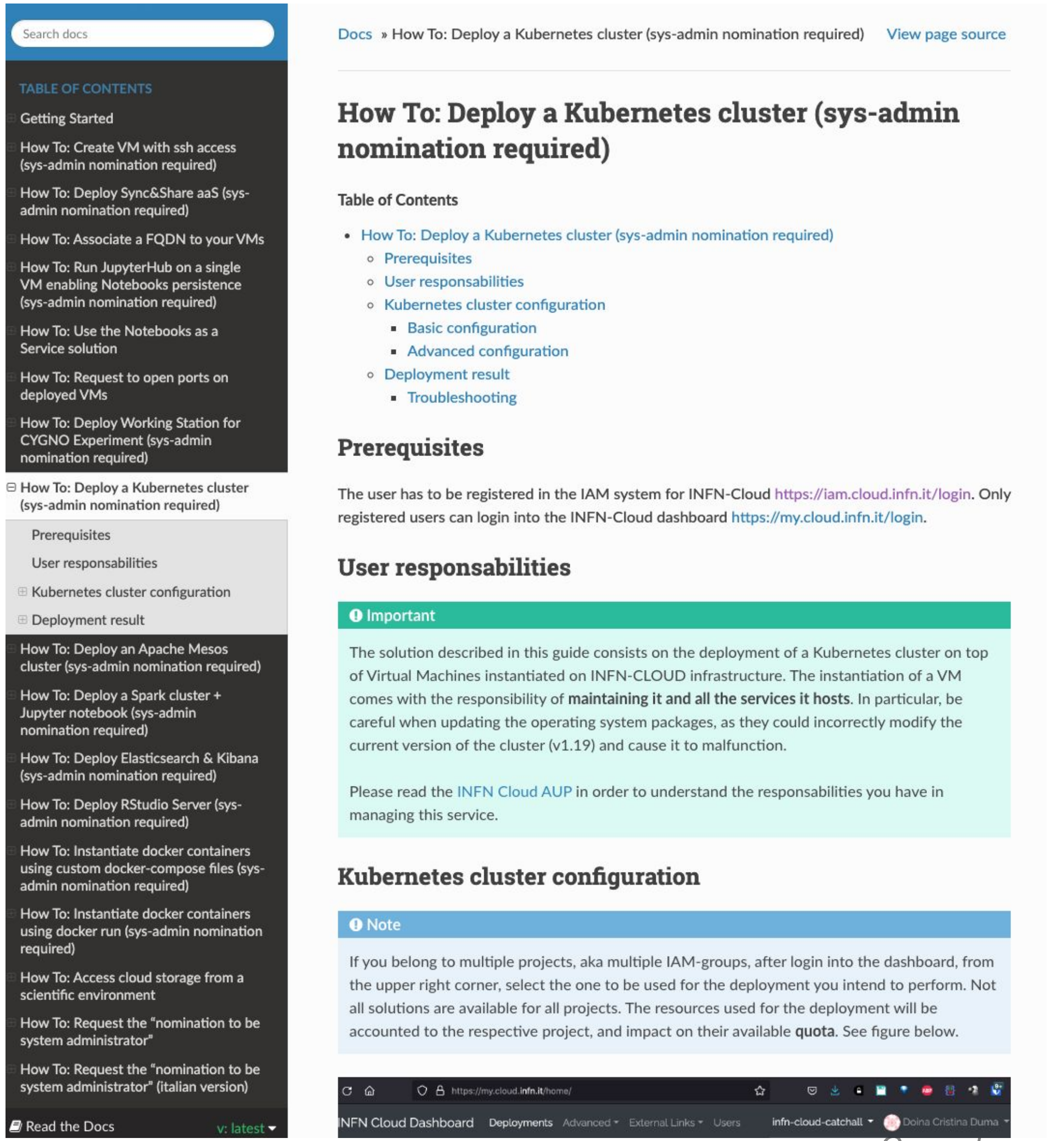
Daemon Sets

Name	Namespace	Images	Labels	Pods	Created
prometheus-node-exporter	prometheus	quay.io/prometheus/node-exporter:v1.3.0	app: prometheus; app.kubernetes.io/managed-by: Helm; chart: prometheus-15.10.1 Show all	1 / 1	a month ago
kube-flannel-ds	kube-system	rancher/mirrored-flannelcni-flannel:v0.18.1	app: flannel; tier: node	2 / 2	a month ago
kube-proxy	kube-system	k8s.gcr.io/kube-proxy:v1.23.8	k8s-app: kube-proxy	2 / 2	a month ago

Deployments

Name	Namespace	Images	Labels	Pods	Created
grafana	grafana	grafana/grafana:8.5.3	app.kubernetes.io/instance: grafana; app.kubernetes.io/managed-by: Helm; app.kubernetes.io/name: grafana Show all	1 / 1	a month ago

How to su guides.cloud.infn.it



Search docs

TABLE OF CONTENTS

Getting Started
How To: Create VM with ssh access (sys-admin nomination required)
How To: Deploy Sync&Share aaS (sys-admin nomination required)
How To: Associate a FQDN to your VMs
How To: Run JupyterHub on a single VM enabling Notebooks persistence (sys-admin nomination required)
How To: Use the Notebooks as a Service solution
How To: Request to open ports on deployed VMs
How To: Deploy Working Station for CYGNO Experiment (sys-admin nomination required)
⊟ How To: Deploy a Kubernetes cluster (sys-admin nomination required)
Prerequisites
User responsabilities
⊞ Kubernetes cluster configuration
⊞ Deployment result
How To: Deploy an Apache Mesos cluster (sys-admin nomination required)
How To: Deploy a Spark cluster + Jupyter notebook (sys-admin nomination required)
How To: Deploy Elasticsearch & Kibana (sys-admin nomination required)
How To: Deploy RStudio Server (sys-admin nomination required)
How To: Instantiate docker containers using custom docker-compose files (sys-admin nomination required)
How To: Instantiate docker containers using docker run (sys-admin nomination required)
How To: Access cloud storage from a scientific environment
How To: Request the "nomination to be system administrator"
How To: Request the "nomination to be system administrator" (italian version)

Read the Docs v: latest

Docs » How To: Deploy a Kubernetes cluster (sys-admin nomination required) View page source

How To: Deploy a Kubernetes cluster (sys-admin nomination required)

Table of Contents

- How To: Deploy a Kubernetes cluster (sys-admin nomination required)
  - Prerequisites
  - User responsabilities
  - Kubernetes cluster configuration
    - Basic configuration
    - Advanced configuration
  - Deployment result
    - Troubleshooting

Prerequisites

The user has to be registered in the IAM system for INFN-Cloud https://iam.cloud.infn.it/login. Only registered users can login into the INFN-Cloud dashboard https://my.cloud.infn.it/login.

User responsabilities

Important

The solution described in this guide consists on the deployment of a Kubernetes cluster on top of Virtual Machines instantiated on INFN-CLOUD infrastructure. The instantiation of a VM comes with the responsibility of **maintaining it and all the services it hosts**. In particular, be careful when updating the operating system packages, as they could incorrectly modify the current version of the cluster (v1.19) and cause it to malfunction.

Please read the INFN Cloud AUP in order to understand the responsabilities you have in managing this service.

Kubernetes cluster configuration

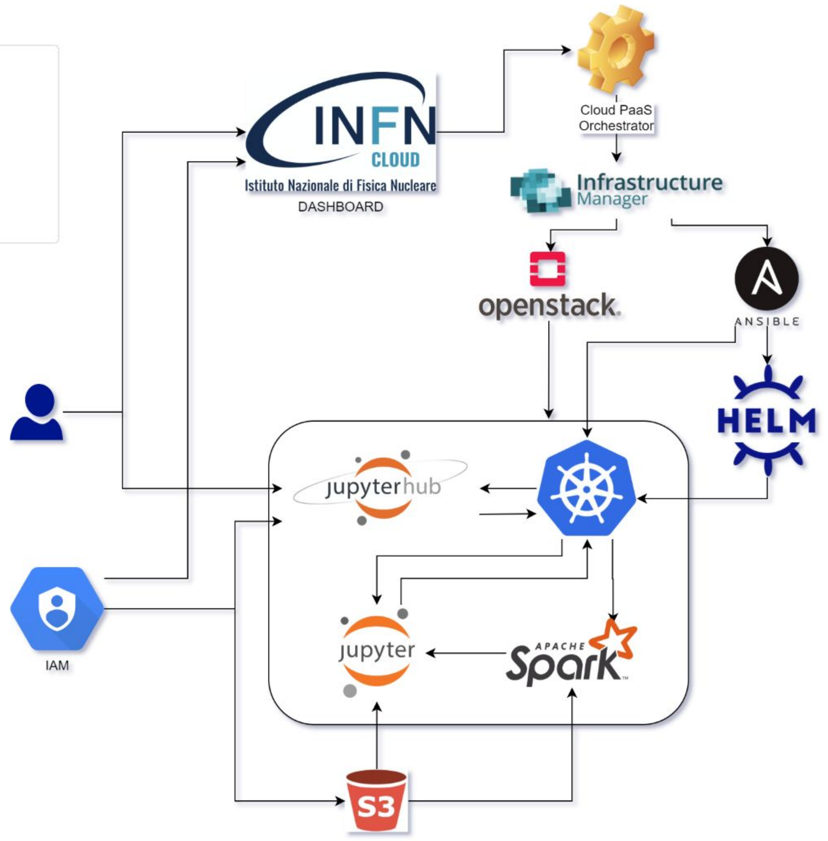
Note

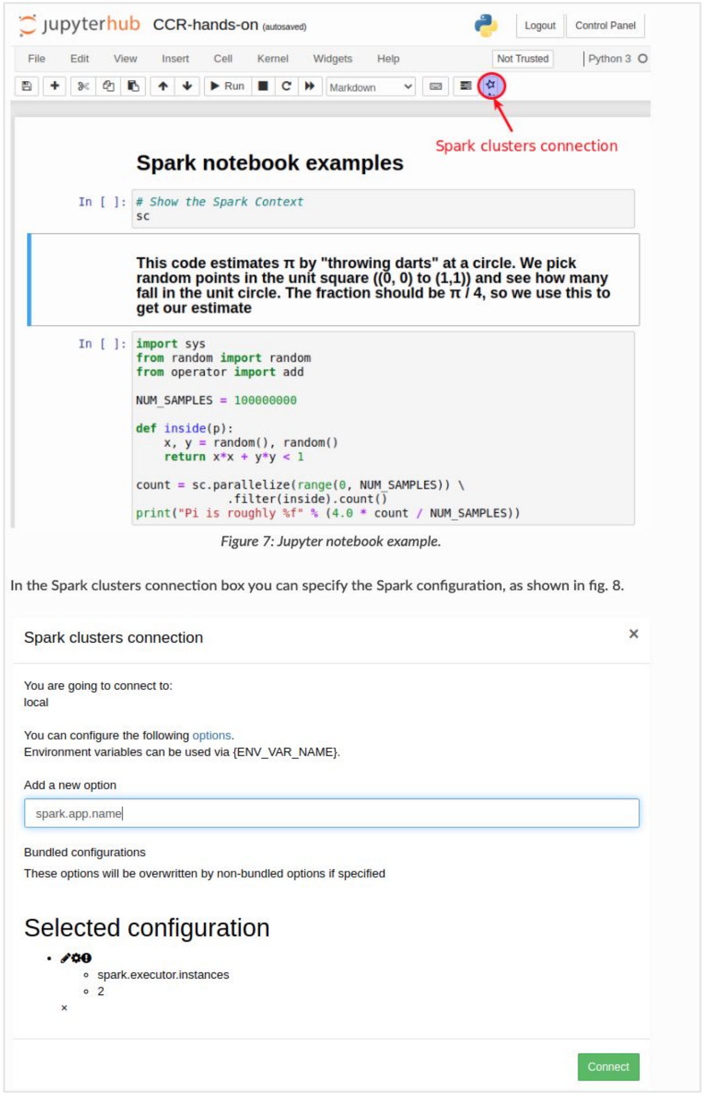
If you belong to multiple projects, aka multiple IAM-groups, after login into the dashboard, from the upper right corner, select the one to be used for the deployment you intend to perform. Not all solutions are available for all projects. The resources used for the deployment will be accounted to the respective project, and impact on their available **quota**. See figure below.

https://guides.cloud.infn.it/docs/users-guides/en/latest/users_guides/howto2.html

Advanced k8s-based services

Jupyter + Spark + K8s




jupyterhub CCR-hands-on (autosaved)

Logout | Control Panel

File Edit View Insert Cell Kernel Widgets Help | Not Trusted | Python 3

Run | Markdown

Spark clusters connection

Spark notebook examples

```
In [ ]: # Show the Spark Context
        sc
```

This code estimates π by "throwing darts" at a circle. We pick random points in the unit square ((0, 0) to (1,1)) and see how many fall in the unit circle. The fraction should be π / 4, so we use this to get our estimate

```
In [ ]: import sys
        from random import random
        from operator import add

        NUM_SAMPLES = 100000000

        def inside(p):
            x, y = random(), random()
            return x*x + y*y < 1

        count = sc.parallelize(range(0, NUM_SAMPLES)) \
                     .filter(inside).count()
        print("Pi is roughly %f" % (4.0 * count / NUM_SAMPLES))
```

Figure 7: Jupyter notebook example.

In the Spark clusters connection box you can specify the Spark configuration, as shown in fig. 8.

Spark clusters connection ×

You are going to connect to:
local

You can configure the following options.
Environment variables can be used via {ENV_VAR_NAME}.

Add a new option

spark.app.name

Bundled configurations

These options will be overwritten by non-bundled options if specified

Selected configuration

-
  - spark.executor.instances
  - 2

×

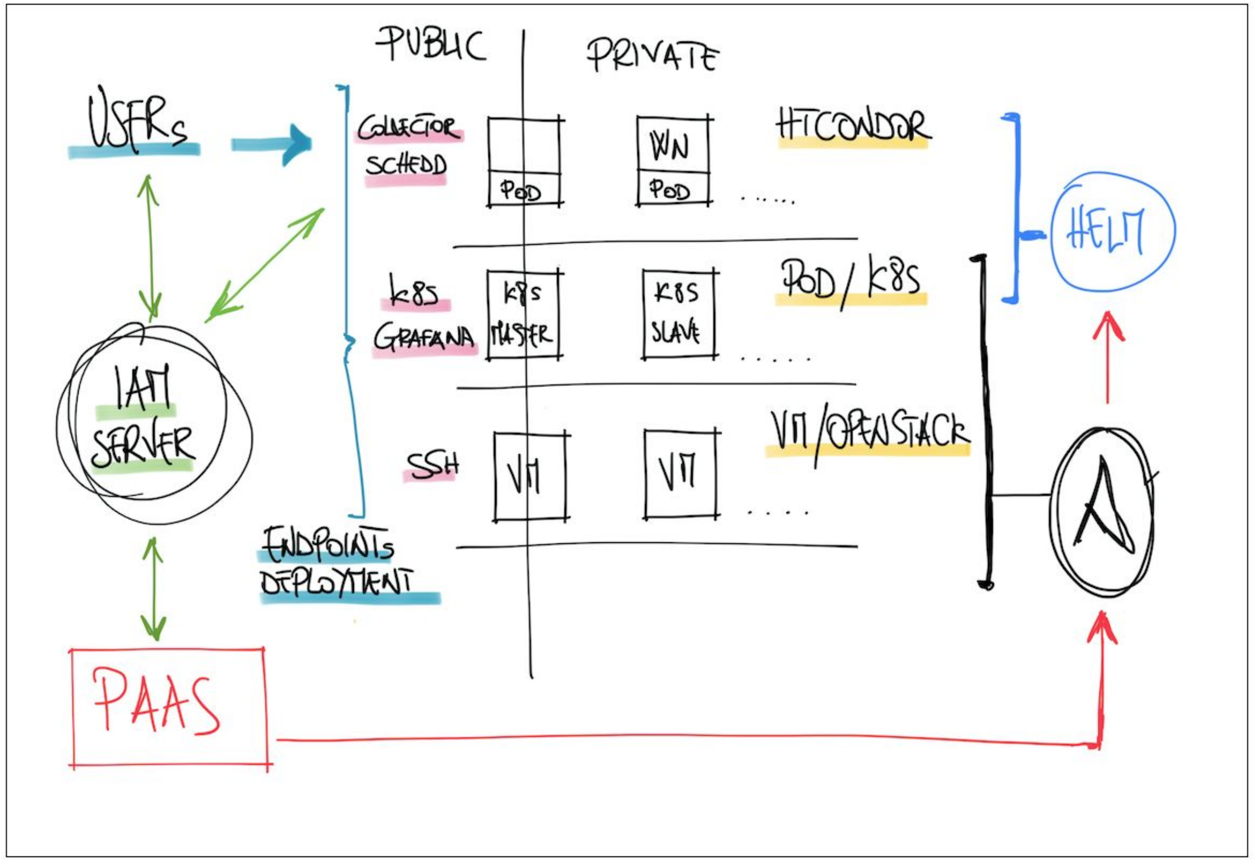
Connect

Advanced k8s-based services (2)

HTCondor + K8s

HTCondor cluster

HTCondor



This deployment instantiate a k8s cluster which is then exploited to automatically deploy a working HTCondor cluster.

The HTCondor cluster deployment is composed by three main components, the CCB, the SCHEDD and the WN, each running on a dedicated POD.

Conclusions

The goal of INFN Cloud is to provide end-users with compute and storage services by offering

- a **portfolio of technical solutions** already developed but extensible – continuously evolving following a **user driven development approach**
- technical support for the end user applications migration to a cloud-based environment
- **transparent** solutions hiding the resources allocation complexity in a **federation of distributed clouds**

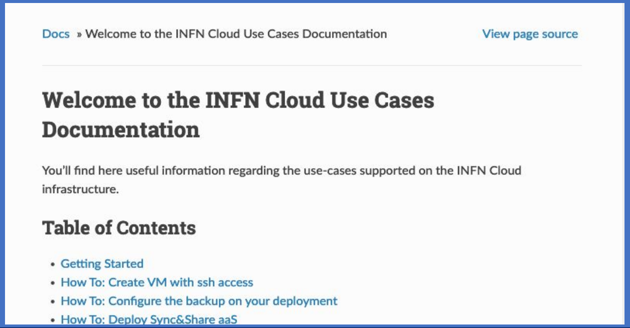
The high-level services shown in this presentation are part of the current portfolio:

- They provide a simple way to run docker containers on cloud resources
- Further (more complex) services have been built starting from these building blocks

If you want to implement a new service or you need to customize an existing one, please contact us at: **cloud-support@infn.it** and you will be redirected to the proper INFN Cloud support team

References

Documentation :
https://guides.cloud.infn.it/docs/users-guides/en/latest/


Docs » Welcome to the INFN Cloud Use Cases Documentation   View page source

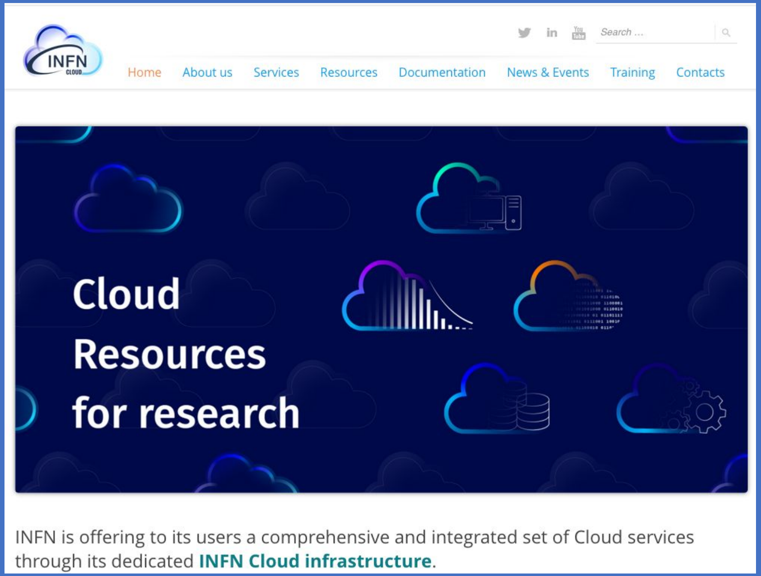
Welcome to the INFN Cloud Use Cases Documentation

You'll find here useful information regarding the use-cases supported on the INFN Cloud infrastructure.

Table of Contents

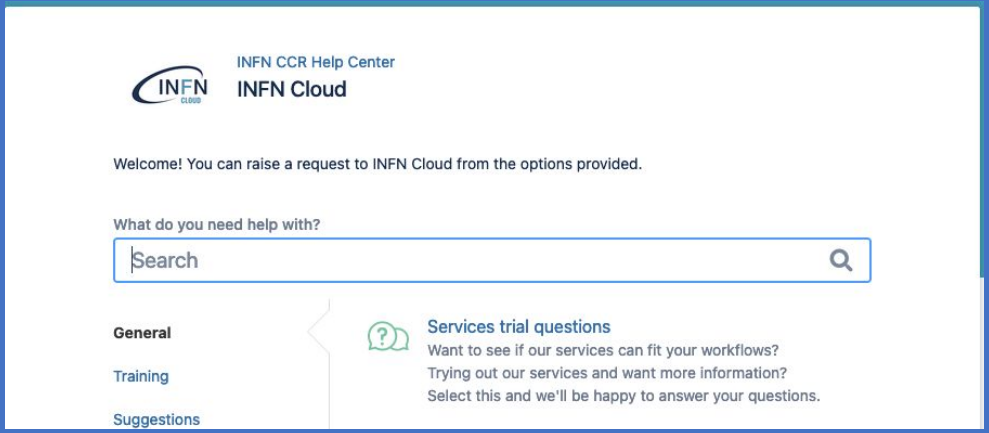
- Getting Started
- How To: Create VM with ssh access
- How To: Configure the backup on your deployment
- How To: Deploy Sync&Share aaS

Web site:
https://www.cloud.infn.it



INFN is offering to its users a comprehensive and integrated set of Cloud services through its dedicated **INFN Cloud infrastructure**.

Support :
https://servicedesk.cloud.infn.it or cloud-support@infn.it


INFN CCR Help Center
INFN Cloud

Welcome! You can raise a request to INFN Cloud from the options provided.

What do you need help with?

Services trial questions
Want to see if our services can fit your workflows?
Trying out our services and want more information?
Select this and we'll be happy to answer your questions.

Thank you

for your attention!

www.cloud.infn.it

For general communications email us at **cloud@lists.infn.it**

To ask for support write to our mailing list **cloud-support@infn.it**, integrated with our ServiceDesk