PROCEEDINGS OF SCIENCE

Leveraging TOSCA orchestration to enable fully automated cloud-based research environments on federated heterogeneous e-infrastructures

M. Antonacci[a,*] **and D. Salomoni**[b] **on behalf of the INFN Cloud Team**

[a] *Istituto Nazionale di Fisica Nucleare,*
Via E. Orabona 4, 70125 Bari, Italy
[b] *Istituto Nazionale di Fisica Nucleare,*
Viale Berti Pichat 6/2, 40127 Bologna, Italy

E-mail: marica.antonacci@ba.infn.it, davide.salomoni@cnaf.infn.it

Cloud computing has offered many opportunities for scientific research, allowing for easy scaling and adapting to new software development methods. However, the lack of integration of existing infrastructures and the consequent fragmentation of resources has hindered the broader adoption of these technologies. The INDIGO-DataCloud project developed solutions for implementing a seamless and transparent access to geographically distributed compute and storage resources, including INDIGO IAM and the INDIGO PaaS [1].

The INFN Cloud infrastructure [2], inaugurated in 2021, is currently using and extending these solutions to provide a wide range of services for scientific communities. These include data analytics and visualization environments, file sync-and-share solutions, and web-based multi-user interactive development environments, among others. The provisioning and configuration of resources are automated through TOSCA templates and Ansible roles and hidden from final users, who can request services through a user-friendly web portal, APIs or a Commnd Line Interface. Security is addressed through consistent authentication and authorization rules, secure configurations, and updates. The platform also allows for deployments on private networks and dedicated VPNs. With its emphasis on ease of use, security, and integration, INFN Cloud represents a significant step forward in supporting scientific research in Italy.

International Symposium on Grids & Clouds (ISGC) 2023 in conjunction with HEPiX Spring 2023 Workshop, ISGC&HEPiX2023
19 - 31 March 2023
Academia Sinica Taipei, Taiwan

*Speaker

PoS(ISGC&HEPiX2023)020

1. Introduction

In recent years, cloud computing has revolutionized scientific research by opening up new opportunities in various fields. With cloud technologies, researchers can now scale applications, adapt quickly, and adopt new software development methods such as *DevOps*, which accelerates time to value. Cloud computing has also made it possible for researchers to analyze large datasets, run complex simulations, and collaborate more effectively across different locations. In general, cloud technologies offer several advantages for scientific research, including:

- Accessibility: Cloud computing makes computing resources more accessible to researchers who may not have access to on-premise computing resources. This can help to democratize access to computing resources and level the playing field for researchers from different locations and backgrounds.
- Scalability: Cloud computing provides researchers with the ability to quickly scale up or down their computing resources as needed. This is particularly beneficial for research projects that require large amounts of computing power for short periods, as researchers can access the resources they need, without having to invest in expensive hardware or worry about managing infrastructure or services.
- Collaboration: Cloud computing makes it easier for researchers to collaborate with colleagues and share data across different locations. Researchers can store their data in the cloud and access it from anywhere with an internet connection, which can help to facilitate collaboration and accelerate the pace of research.
- Speed: Cloud computing allows researchers to quickly spin up new environments and test new applications or software. This can help to accelerate the time to market for new research findings and innovations.
- Security: Cloud providers typically have robust security measures in place to protect data, which can provide researchers with greater peace of mind. Additionally, cloud providers can offer disaster recovery and backup services to help protect against data loss.

However, the lack of integration of existing infrastructures and the fragmentation of resources still pose significant challenges to the broader adoption of cloud computing in scientific research. Another challenge is the technical complexity of cloud computing, which can be a barrier to adoption for researchers who may not have a strong background in IT. Cloud computing involves a range of technical concepts, such as virtualization, networking, and security, that can be difficult to understand for researchers who are not familiar with these areas. This can make it challenging for researchers to get started with cloud computing, and may cause some to avoid using cloud technologies altogether. In addition to technical complexity, some researchers may be resistant to adopting cloud computing because they are comfortable with existing infrastructures and methods. This can create a cultural barrier to adoption that must be overcome through education and training.

INFN has recently undergone a restructuring of its internal organization and solutions to better support computing and storage resources and services. One of the first outcomes of this effort is the creation of INFN Cloud, a federated cloud infrastructure centered around a data

lake architecture. Drawing on INFN's experience with Grid and Cloud computing solutions for scientific communities, INFN Cloud provides a customizable service portfolio to meet the needs of multidisciplinary scientific communities. The portfolio includes traditional IaaS offerings, as well as more sophisticated PaaS and SaaS solutions that are tailored to the requirements of specific communities. In addition to its technical capabilities, INFN Cloud also places special attention on training and on assisting user communities to migrate their use cases onto the cloud. INFN Cloud provides resources and support to help users learn how to use the platform effectively and efficiently. This includes training programs and online resources that are tailored to the needs of different user communities.

2. The INFN Cloud Platform

For several decades, INFN has provided and supported the largest research and academic distributed infrastructure in Italy. This infrastructure includes a large national data center (CNAF in Bologna) and nine other sizable data centers, all interconnected through the high-capacity GARR network [3]. The infrastructure utilizes Grid or Cloud protocols to serve the needs of international collaborations in physics and other scientific domains. Recently, INFN has expanded its offerings to include a comprehensive and integrated set of Cloud services through the dedicated INFN Cloud infrastructure.

The main goal of INFN Cloud is to address the challenges faced by researchers in accessing and utilizing distributed compute and storage resources, at the same time offering new opportunities to better perform scientific tasks, in a shorter time frame. As described in the previous section, the most common challenges include technical complexity, lack of integration of existing infrastructures, and security concerns. INFN Cloud aims to provide a portfolio of high-level services that can be easily accessed by INFN researchers, leveraging the distributed compute and storage resources available across Italy. By offering a flexible and scalable platform that is designed according to user requirements, INFN Cloud enables researchers to take advantage of the benefits of Cloud computing while also addressing the challenges that have historically hindered the adoption of Cloud technologies in scientific research.

INFN Cloud is built on a foundation of proven, key architectural elements. One of them is the emphasis on an open-source, vendor-neutral architecture, which ensures that the platform is not tied to any specific vendor or technology. Another important element is the flexible federation of existing cloud infrastructures for both compute and data, which enables researchers to access a wide range of resources across Italy. Dynamic orchestration of resources is achieved via the INDIGO PaaS Orchestrator [1], which allows for the automated deployment and management of cloud resources. Finally, consistent authentication and authorization mechanisms are ensured at all cloud levels via INDIGO-IAM [4], a federated solution for authentication and authorization that is fully compliant with European Open Science Cloud (EOSC) and industry standards.

From an infrastructural perspective, INFN Cloud is built on a core Backbone, realized through an OpenStack installation with two regions, connecting the large data centers of CNAF and Bari (Figure 1). These two data centers provide the primary compute and storage resources for the platform.

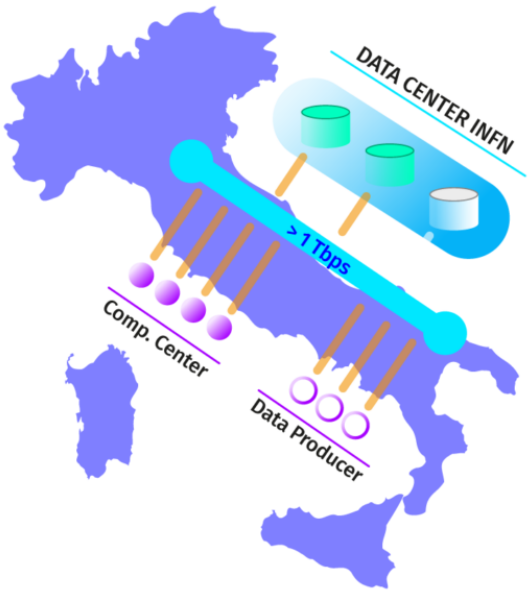


Figure 1: INFN Cloud is designed as a federation of pre-existing infrastructures.

In addition to these core data centers, other satellite sites spread across Italy are loosely coupled to the Backbone by means of the federation layer based on the INDIGO PaaS Orchestrator. These sites provide additional compute and storage resources that can be used to support scientific use cases. New federated sites can be connected at any time and can differ in terms of the cloud middleware utilized (such as OpenStack, OpenNebula, Kubernetes, etc.) and the version of the middleware. These sites can also differ in the storage and compute offerings they provide, with some offering specialized hardware or high-performance storage, for example. The heterogeneity among federated sites enables researchers to access a diverse set of resources across Italy and facilitates the deployment of a wide range of scientific research use cases on the INFN Cloud platform.

The core services of the INFN Cloud, as well as some centralised, fully managed, high-level services, are hosted on the Backbone, leveraging its high-availability and disaster recovery capabilities (data are replicated on the two regions) to ensure that critical services are always available and operating at peak efficiency.

The federated architecture of the INFN Cloud enables researchers to access a wide range of resources across Italy, without being tied to any specific location or infrastructure. This provides a high degree of flexibility and scalability, which is essential for supporting scientific research in a rapidly changing environment. Overall, the design of the INFN Cloud infrastructure is intended to provide a high degree of reliability, performance, and security, while also being flexible enough to support a wide range of scientific research use cases.

The INFN Cloud portfolio, available via an easy-to-use web interface, is designed according to clear users' requirements. It is based on composable, open-source solutions and can be easily extended by the INFN Cloud support team or directly by end users.

3. The Federation Middleware

The INFN Cloud platform's federation middleware is based on the INDIGO PaaS Orchestration system, a collection of open-source microservices that implement an abstraction layer enabling seamless access to compute and storage resources made available by various and heterogeneous providers. These providers can include public and private clouds (OpenStack, OpenNebula, AWS, Azure, etc.), container orchestration platforms such as Kubernetes and Apache Mesos, and other types of infrastructures. The PaaS Orchestration system provides smart scheduling functionalities that allow the automatic selection of the optimum provider based on compute and storage requirements vs provider capabilities.

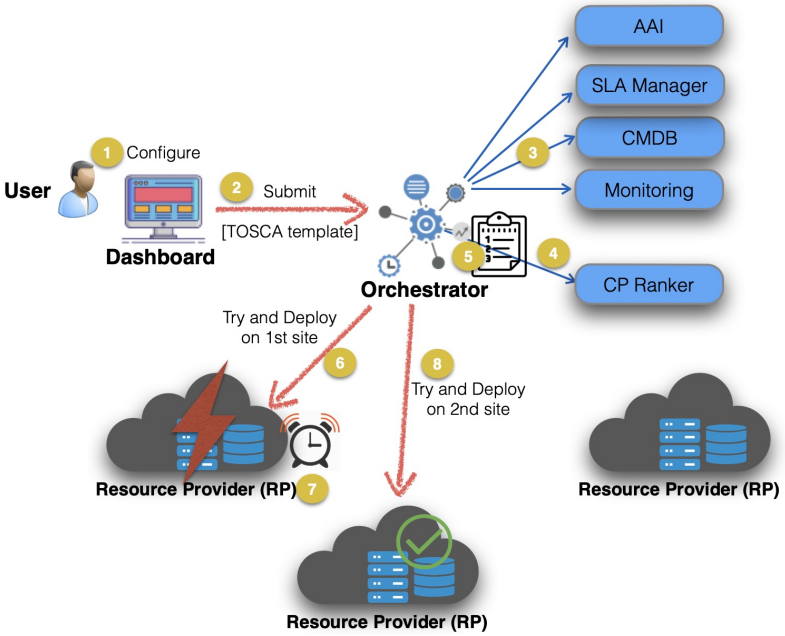


Figure 2: Hih-level architecture and workflow of the PaaS Orchestration system.

This means that when a user submits a deployment request (see Figure 2), the Orchestrator chooses the best provider from a list of potential providers based on several criteria such as resource quotas, monitoring data, support for specialized hardware, and data location, among others. The deployment request is described through the TOSCA language [5], which is a standard for describing cloud applications and services. To make this selection, the Orchestrator interacts with auxiliary services such as the Service Level Agreement Manager (SLAM), the Configuration Management DataBase (CMDB), and the Monitoring System to obtain information about the available providers and their capabilities. The Cloud Provider Ranker (CPR) then uses this information to generate a

sorted list of candidate providers for the deployment. The best provider is at the top of the list, and the deployment request will be routed to it as the first trial. In the event of a failure, a retry strategy is implemented, and the deployment is rescheduled on the next available provider in the list.

Client interfaces are offered for advanced users, including REST APIs, CLI, and Python bindings, which enable them to interact with the platform programmatically. Additionally, a user-friendly web dashboard is available for end-users who do not have specialized skills.

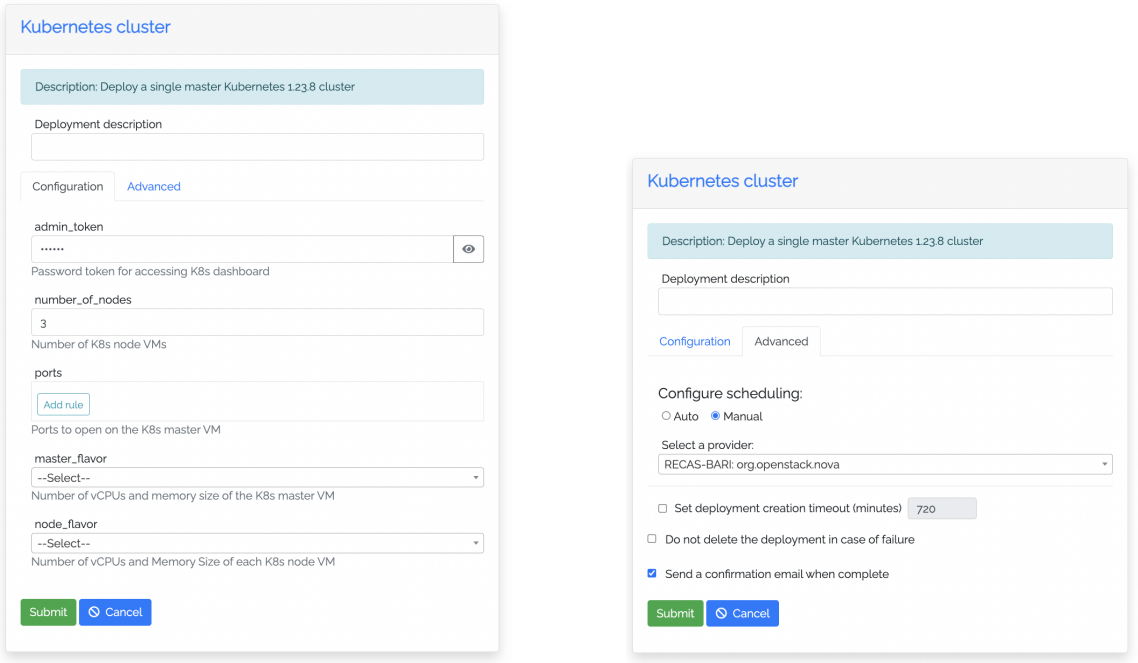
Unlike other interfaces that require a deep understanding of the underlying technical details, the PaaS dashboard abstracts away all of the technical complexities and provides an intuitive interface for managing service deployments. This means that users do not need to have any TOSCA knowledge to use the platform effectively. One of the features of the dashboard is OpenID-Connect [6] Authentication through the INFN Cloud IAM Service (a dedicated instance of INDIGO IAM), which provides a secure and standardized means of accessing the platform. This feature ensures that only authorized users can access the platform and its resources. Another key feature of the PaaS Dashboard is multi-tenancy, which enables multiple users or organizations to share the same infrastructure while keeping their data and applications separate. This feature ensures that users can deploy and manage their services independently, without interfering with others using the same infrastructure. The dashboard also provides a dynamic view of the service catalog, which is dependent on the user group membership. This feature enables users to view and access only the services that are relevant to their particular group or role, ensuring that they can easily find and use the services that are most important to them. Finally, the dashboard provides a feature for secrets management, which is achieved through Hashicorp Vault [7] integration. This feature allows users to securely store and manage sensitive information, such as passwords and ssh keys, ensuring that this information is kept confidential and only accessible to authorized users.

For each service in the catalog, the dashboard automatically generates a configuration form in HTML based on the TOSCA template that describes the service. This form enables users to customize the deployment of the service to meet their specific needs. The configuration form presents a range of options that users can select to configure their service deployment. These options can include (Figure 3a) the number of virtual nodes, their size (cores and memory), the type of storage to use, and other parameters that are relevant to the service being deployed. The form also allows users to choose the scheduling strategy (Figure 3b) that will be used for their deployment:

- **automatic**, which lets the Orchestrator select the best provider based on compute and storage requirements versus provider capabilities;
- **manual**, which allows users to choose the provider from a drop-down menu that is automatically created by the Dashboard with the list of providers returned by the SLA Manager service.

The service implementation strategy used in the INFN Cloud platform relies on the Infrastructure as Code (IaC) paradigm. Iac allows users to define "What" they need, rather than specifying "How" a particular service should be implemented. This approach enables a Lego-like construction of services and promotes the reusability of modules to create the desired infrastructure and services.

To implement this approach, the TOSCA language is used to model the topology of the whole application stack. TOSCA is a standard for describing cloud applications and services, and provides



(a) Basic configuration: the user can customize the deployment specifying the nodes' number and flavor, the ports to be opened, and so on.

(b) Advanced configuration: the user can select the scheduling strategy for the deployment and the overall creation timeout.

Figure 3: Example of configuration form for requesting a Kubernetes cluster.

a structured way to describe the components and relationships of a service. The TOSCA templates define the high-level service architecture, including the compute, storage, and networking resources required for the service.

Once the TOSCA templates are defined, Ansible [8] is used to automate the configuration of the virtual environments. Ansible is an open-source automation tool that simplifies the deployment and management of applications and services. It provides a simple syntax for defining the desired state of a system, and it can be used to automate the configuration of virtual environments, including the installation and configuration of software packages and dependencies.

Finally, Docker [9] is used to encapsulate the high-level application software and runtime. Docker is a containerization technology that enables the creation of lightweight, portable, and isolated environments for running applications. By encapsulating the application software and runtime in Docker containers, users can ensure that their software will run consistently across different environments.

Overall, the use of the IaC paradigm, TOSCA, Ansible, and Docker provides a powerful toolset for implementing services on the INFN Cloud platform. This approach promotes modularity, reusability, and consistency, and it enables users to focus on their research and scientific work, rather than spending time on infrastructure management.

4. The Service Catalog

The INFN Cloud service portfolio is continuously updated and expanded to meet the evolving needs of users.

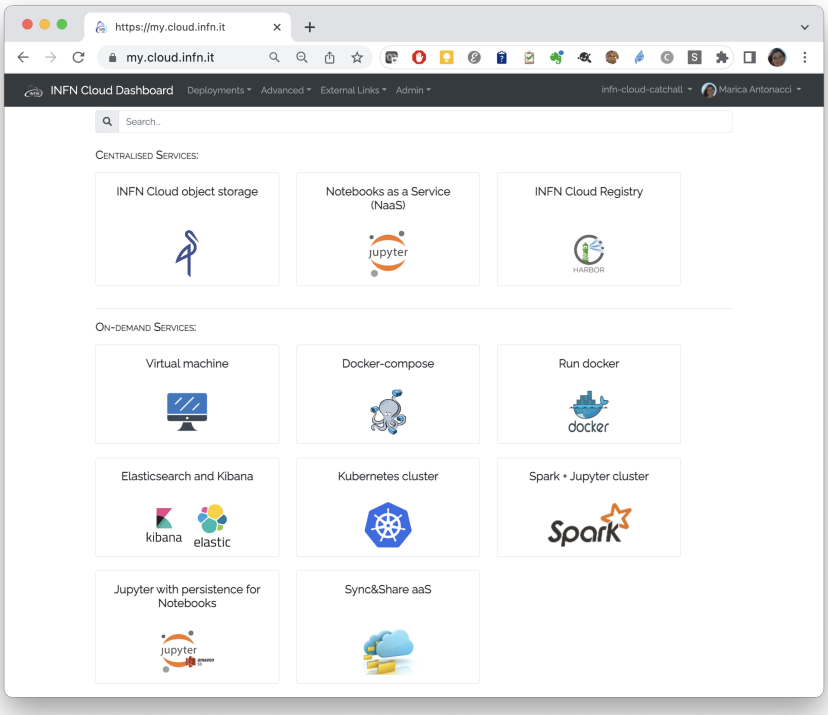


Figure 4: Service Catalog. Both centrally managed services and on-demand servives can be accessed through the dashboard.

The portfolio (Figure 4) includes a range of *general-purpose* services, such as:

- Virtual Machines: Users can deploy virtual machines with or without external block storage, and the option to add a Docker engine and docker-compose. This enables users to run dockerized services automatically on top of the virtual machines.
- Data Analytics and Visualization Environments: INFN Cloud provides data analytics and visualization environments based on Elasticsearch [10] and Kibana [11], enabling users to analyze and visualize their data with ease.
- File Sync & Share Solution: INFN Cloud offers a file sync and share solution based on ownCloud [12] or NextCloud [13] with several features, including:
  - Replicated backend storage on the S3-compliant Object Storage provided by the INFN Cloud Backbone.
  - Automatic configuration for enabling INDIGO IAM OpenID Connect authentication.

- Pre-installed and configured backup cron jobs for safely storing configuration and metadata on the Object Storage for future restore in case of disaster.
- Integrated application and backup monitoring based on Nagios [14].

In addition to the general-purpose services, INFN Cloud also offers on-demand interactive analysis environments. These environments are web-based development environments built on JupyterLab [15], enabling users to engage in interactive development using notebooks, code, and data. The environments include persistent storage areas for storing results and notebooks, as well as a monitoring system based on Prometheus [16] and Grafana [17]. What's more, the interactive analysis environments are customizable based on the specific needs of each experiment. Users can customize the environment with experiment-specific extensions, libraries, and configurations to suit their particular requirements. For instance, the Cygno experiment [18] for Dark Matter direct detection is supported by INFN Cloud with customized Python/ROOT kernels, pre-installed libraries for event reconstruction, data analysis, and simulation based on GEANT4 [19] and Garfield++ [20] software. Additionally, the Cygno experiment can make use of CVMFS [21] mounts to access the necessary software and libraries. Another customization is available for the ML-INFN Project, an INFN-funded project that aims to lower the potential barriers for accessing specialized hardware for the exploitation of Machine Learning techniques. In this case, the JupyterLab instances are designed to access one or more GPUs automatically as the needed drivers and configurations are managed by the system. Moreover, GPU partitioning based on the nvidia MIG feature [22] is supported for optimal utilization, enabling users to take advantage of specialized hardware for their ML workloads. This flexibility ensures that users can work with the tools and frameworks they are most comfortable with, while taking advantage of the powerful infrastructure provided by INFN Cloud.

Moreover, the service portfolio offers the possibility to deploy Kubernetes clusters on demand, providing Kubernetes as a Service (KaaS). With KaaS, users can quickly and easily deploy and manage their own Kubernetes clusters, without the need for manual installation and configuration.

In addition to KaaS, INFN Cloud provides more advanced and complex services that exploit the advantages and capabilities of Kubernetes. In particular, services such as Spark, integrated with JupyterHub, and HTCondor cluster on demand are deployed on top of a Kubernetes cluster. From a technical standpoint, the deployment of services based on Kubernetes extends the templating and encapsulation logic discussed earlier. This demonstrates the power and versatility of the original model while maintaining the declarative approach, thanks to the adoption of Helm Charts [23]. Helm Charts allow for the definition, installation, and upgrading of Kubernetes applications while still adhering to a declarative syntax.

INFN Cloud provides both on-demand and centrally managed services. The main difference between these two types of services is the level of responsibility for maintenance and daily operations. In the case of on-demand services, users are responsible for administering the service, including updates and daily operations. This provides users with greater flexibility and control over their services, enabling them to customize their deployments to meet their specific needs.

On the other hand, with centrally managed services, the INFN Cloud team takes care of maintenance and daily operations, providing users with a hassle-free experience. These services are accessible through a Software as a Service (SaaS) model, where users access the service through

a web-based interface without needing to worry about the underlying infrastructure. Currently the following services are delivered in SaaS mode:

- INFN Cloud Object Storage: INFN Cloud provides a multi-region OpenStack Swift-based Object Storage, utilizing the Backbone created between the Bari and CNAF data centers. Users can access the Object Storage through S3 APIs and a web interface via the MinIO Gateway [24]. Access control rules are defined using Open Policy Agent (OPA) [25], based on IAM token claims.

- Notebook as a Service (NaaS): This service provides a JupyterHub backed by Kubernetes clusters hosted on the INFN Cloud Backbone. The service offers high availability and failover mechanisms across the two sites. Each notebook is connected with the Object Storage for data persistence.

- INFN Cloud Registry: INFN Cloud offers a centrally managed registry, based on the Harbor software [26]. The registry provides users with a secure and reliable platform for storing and managing container images, enabling them to deploy their applications with ease.

5. Securing scientific research environments

Security is a critical aspect of any cloud computing environment, particularly in the context of scientific research where sensitive data is often involved. To address the growing need for enhanced data security, legal compliance, and ethical requirements, measures must be taken to improve the security of deployed services and INFN Cloud has implemented several measures.

Firstly, the PaaS Orchestration system has been extended to enable the deployment of Virtual Machines (VMs) on private networks. This involves contextualization via a "jump host" (Figure 5), which ensures that the VMs are isolated from the public network. Additionally, a Vault instance is provided for storing sensitive information, ensuring that data is kept secure and confidential.

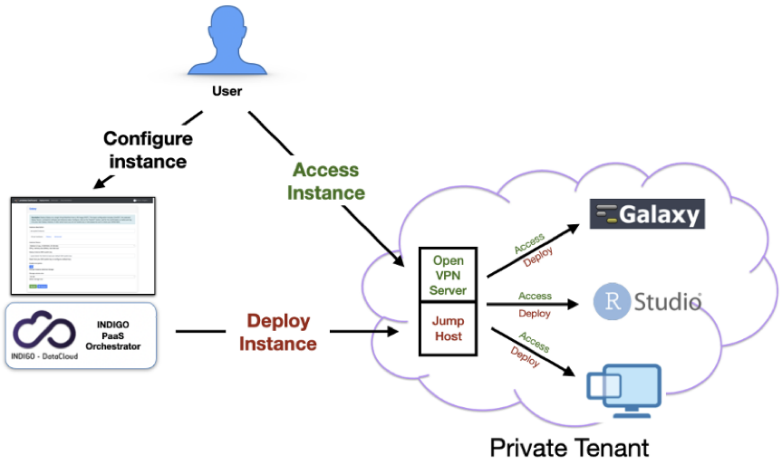


Figure 5: Deployment of virtual environments on isolated private networks.

In this scenario, users are required to access the deployed services through a Virtual Private Network (VPN): to ensure authentication through the same IAM used for accessing the rest of the INFN Cloud infrastructure services, a PAM (Pluggable Authentication Module) module [27] has been developed.

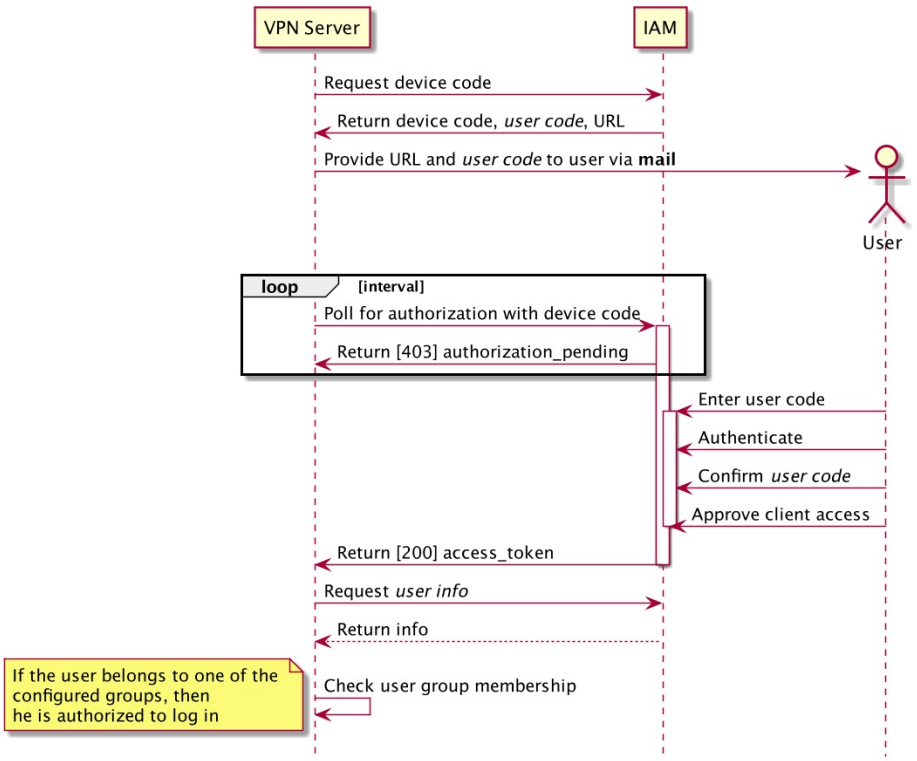


Figure 6: VPN authentication integrated with IAM. The developed PAM module [27] allows to login using OpenID Connect credentials. It uses the OAuth2 Device Flow, which means that during the login process, the user receives an email with a link to log in to IAM, which is in charge to authenticate the user. The PAM module then checks if the user belongs to the right group(s), if any has been specified in the module configuration, and allows or denies access.

This PAM module is installed on the VPN server (Figure 6) and enables authentication using the device code authorization flow; moreover, group-based authorization is supported and can be easily enabled as one of the available configuration options of the module. This approach ensures that only authorized users can access the deployed services, while maintaining a consistent authentication mechanism across the entire INFN Cloud infrastructure. The reference implementation is built upon OpenVPN [28], a widely used open-source VPN solution.

In addition to the security measures discussed earlier, INFN Cloud has developed an automatic testing system to ensure that service templates are regularly updated with security patches. The system uses Jenkins [29], an open-source automation server, to evaluate TOSCA templates through pre-defined, fully automated job pipelines. These pipelines conduct automatic checks for every service in the catalog, including security vulnerability scans, to ensure that the deployed services

are secure and reliable.

6. Conclusions

INFN Cloud aims to address the new challenges of computing for scientific research providing researchers with simplified access to distributed resources, eliminating the need for specialized IT knowledge. The platform is designed to facilitate complex analyses and streamline the research process by enabling the deployment of complex services with minimal effort, allowing researchers to focus on their work.

Moving forward, we have identified two key areas of focus for further development. Firstly, we will work towards improving the scheduling algorithm to guarantee that computational resources are allocated as efficiently as possible, taking into account also data access patterns. This will improve the overall performance of the platform. Secondly, we plan to provide integrated solutions for data management across the INFN Cloud distributed sites. This will enable researchers to seamlessly access and manage data across multiple sites, further streamlining the research process and enhancing collaboration among teams.

Overall, the INFN Cloud infrastructure represents a significant expansion of the services offered by INFN. By providing an integrated set of cloud services that are designed according to user requirements, INFN Cloud enables users to take advantage of the benefits of Cloud computing while also leveraging the experience and expertise of INFN in supporting scientific research. With the ability to extend to other cloud providers, INFN Cloud provides a cutting-edge, flexible and scalable platform for scientific research in Italy and beyond.

References

[1] Salomoni, D., Campos, I., Gaido, L. et al. *INDIGO-DataCloud: a Platform to Facilitate Seamless Access to E-Infrastructures*. *J Grid Computing* **16**, 381–408 (2018). https://doi.org/10.1007/s10723-018-9453-3

[2] INFN Cloud. *https://www.cloud.infn.it*

[3] GARR Network, accessed 17 April 2023, https://www.garr.it/en/infrastructures/network-infrastructure/our-network

[4] A Ceccanti et al. *The INDIGO-Datacloud Authentication and Authorization Infrastructure*. 2017 *J. Phys.: Conf. Ser.* **898** 102016 https://doi.org/10.1088/1742-6596/898/10/102016

[5] OASIS: TOSCA 1.0 (Topology and Orchestration Specification for Cloud Applications), Version 1.0. http://docs.oasis-open.org/tosca/TOSCA/v1.0/TOSCAv1.0.pdf (2013)

[6] OpenID Connect, accessed 17 April 2023, https://openid.net/connect/

[7] Vault by HashiCorp, accessed 17 April 2023, https://www.vaultproject.io/

[8] Ansible, accessed 17 April 2023, https://docs.ansible.com/ansible/latest/index.html

[9] Docker: Accelerated, Containerized Application Development, accessed 17 April 2023, https://www.docker.com/

[10] Elasticsearch, accessed 17 April 2023, https://www.elastic.co/what-is/elasticsearch

[11] Kibana, accessed 17 April 2023, https://www.elastic.co/what-is/kibana

[12] ownCloud - share files and folders, easy and secure, accessed 17 April 2023, https://owncloud.com/

[13] Nextcloud - Online collaboration platform, accessed 17 April 2023, https://nextcloud.com/

[14] Nagios, accessed 17 April 2023, https://www.nagios.org/

[15] Project Jupyter, accessed 17 April 2023, https://jupyter.org/

[16] Prometheus - Monitoring system & time series database, accessed 17 April 2023, https://prometheus.io/

[17] Grafana: The open observability platform | Grafana Labs, accessed 17 April 2023, https://grafana.com/

[18] Amaro, F.D.; et al. The CYGNO Experiment. Instruments 2022, 6, 6. https://doi.org/10.3390/instruments6010006

[19] Geant4 - CERN, accessed 17 April 2023, https://geant4.web.cern.ch/

[20] Garfield++, accessed 17 April 2023, https://www.desy.de/ zenker/FLC/garfieldpp.html

[21] CernVM File System, accessed 17 April 2023, https://cernvm.cern.ch/fs/

[22] Multi-Instance GPU (MIG), accessed 17 April 2023, https://www.nvidia.com/en-us/technologies/multi-instance-gpu/

[23] Helm - The package manager for Kubernetes, accessed 17 April 2023, https://helm.sh/

[24] MinIO | High Performance, Kubernetes Native Object Storage, accessed 17 April 2023, https://min.io/

[25] Open Policy Agent, accessed 17 April 2023, https://www.openpolicyagent.org/

[26] Harbor, accessed 17 April 2023, https://www.openpolicyagent.org/

[27] PAM module for OAuth 2.0 Device flow, github repo: https://github.com/maricaantonacci/pam_oauth2_device

[28] Open Source OpenVPN, accessed 17 April 2023, https://community.openvpn.net/openvpn

[29] Jenkins, accessed 17 April 2023, https://www.jenkins.io/