**VIETNAM NATIONAL UNIVERSITY HO CHI MINH CITY**

**UNIVERSITY OF INFORMATION TECHNOLOGY**

**COMPUTER NETWORKS AND DATA COMMUNICATION**

**FINAL PROJECT – GROUP 11**

**Distributed computing system NT533.N21.MMCL**

**Project: Building OpenStack Yoga with Ubuntu Server 20.04 LTS**

**Lecturer: Bui Thanh Binh**

**Members:**

| Le Thi Thanh Nhi | 19521964 |
| --- | --- |
| Duong Duy Truong | 20522080 |
| Nguyen Le Dac Hiep | 20521312 |
|  |  |

**HO CHI MINH CITY, 2023**

**TABLE OF CONTENTS**

**1**[**.**](#_heading=h.tyjcwt) **RESEARCH AND ANALYSE 4**

[**2.1.**](#_heading=h.3dy6vkm) **Research** **about component:** 4

[**2.2.**](#_heading=h.4d34og8) **Analyze how components interact with each other** 5

[**2.3.**](#_heading=h.3rdcrjn) **Advantages and disadvantages** 6

**2**[**.**](#_heading=h.1fob9te) **DEPLOYMENT MODEL AND CONDITIONS** [**8**](#_heading=h.3rdcrjn)

[**3.**](#_heading=h.26in1rg) **CONCLUSION 9**

[**3.1 – Result** 9](#_heading=h.lnxbz9)

**TABLE OF IMAGES**

[Image 1. Deployment model 4](#_heading=h.3znysh7)

[Image 2. Set up environment 4](#_heading=h.2et92p0)

[Image 3. About component of Openstack 5](#_heading=h.1t3h5sf)

[Image 4. Components interact with each other 6](#_heading=h.2s8eyo1)

[Image 5. Core services 8](#_heading=h.17dp8vu)

[Image 6. Login admin 10](#_heading=h.35nkun2)

[Image 7. Admin interface 10](#_heading=h.1ksv4uv)

[Image 8. Usage statistics 11](#_heading=h.44sinio)

[Image 9. API Access 11](#_heading=h.2jxsxqh)

[Image 10. Instances statistics of Project 12](#_heading=h.z337ya)

[Image 11. Images statistics of Project 12](#_heading=h.3j2qqm3)

[Image 14. Networks statistics 12](#_heading=h.1y810tw)

[Image 15. Routers statistics 13](#_heading=h.4i7ojhp)

[Image 16. Security groups statistics 13](#_heading=h.2xcytpi)

[Image 17. Floating IPs 14](#_heading=h.1ci93xb)

[Image 18. Overview 14](#_heading=h.3whwml4)

[Image 19. All hypervisors 15](#_heading=h.2bn6wsx)

[Image 20. Host aggregates 15](#_heading=h.qsh70q)

[Image 21. Instances statistics of Admin 16](#_heading=h.3as4poj)

[Image 22. Images statistics of Admin 16](#_heading=h.1pxezwc)

[Image 23. Flavors statistics of Admin 17](#_heading=h.49x2ik5)

[Image 24. Identity of Project 17](#_heading=h.2p2csry)

[Image 25. Ping from instance to controller node 18](#_heading=h.147n2zr)

[Image 26. Ping from instance to compute node 18](#_heading=h.3o7alnk)

[Image 27.](#_heading=h.23ckvvd) Ping to a location in network topology [19](#_heading=h.23ckvvd)

# **RESEARCH AND ANALYSE**

* 1. **Research** **about component:**

The components of OpenStack are open source software projects that provide basic cloud services such as computing, storage, networking, authentication and interface. Some of the main components of OpenStack are:

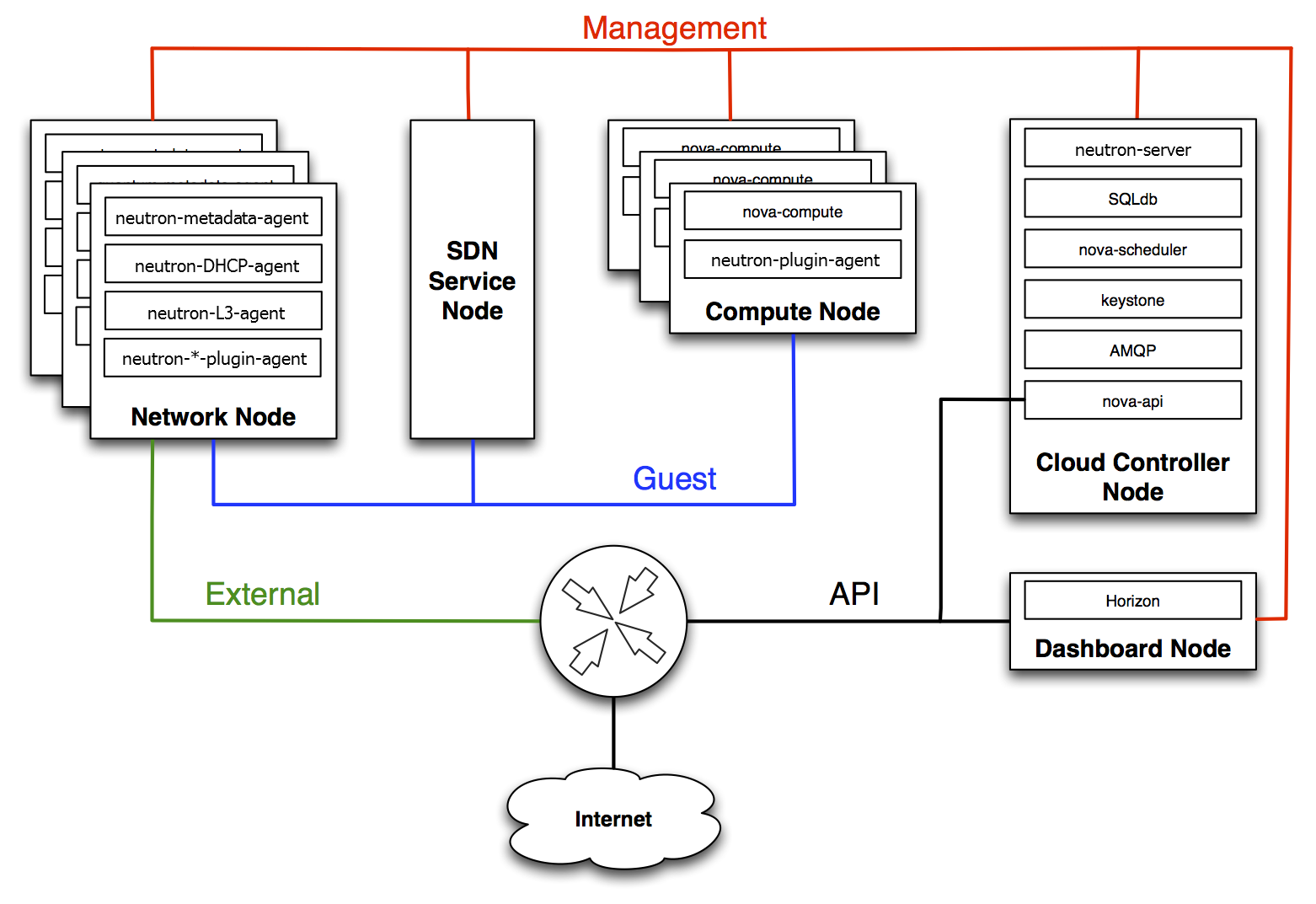
* Compute (Nova): This component provides virtual machines (VMs) on-demand, allowing users to launch and manage instances. It handles the automated provisioning and lifecycle management of VMs.
* Networking (Neutron): Neutron manages the networking aspects of the cloud infrastructure, including creating and managing virtual networks, routers, and security groups. It enables users to define and configure network connectivity between instances and external networks.
* Storage (Cinder and Swift): OpenStack offers two storage options. Cinder provides block storage, similar to traditional hard drives, which can be attached to instances for persistent storage. Swift is an object storage system that allows users to store and retrieve large amounts of unstructured data.
* Identity (Keystone): Keystone is responsible for authentication and authorization services in OpenStack. It provides a centralized authentication system that integrates with existing user directories and supports various authentication mechanisms.
* Image service (Glance): Glance manages virtual machine images used to create instances. It allows users to discover, register, and retrieve images to boot instances.
* Orchestration (Heat): Heat is an orchestration engine that enables users to describe and deploy complex cloud applications using templates. It automates the process of provisioning resources and managing dependencies.
* Dashboard (Horizon): Horizon provides a web-based graphical user interface (GUI) for managing and provisioning resources in an OpenStack cloud. It offers an intuitive interface for administrators and end-users to interact with OpenStack services.
* A screenshot of a computer

  Description automatically generated with medium confidence

*Image 3. About component of Openstack*

* 1. **Analyze how components interact with each other**

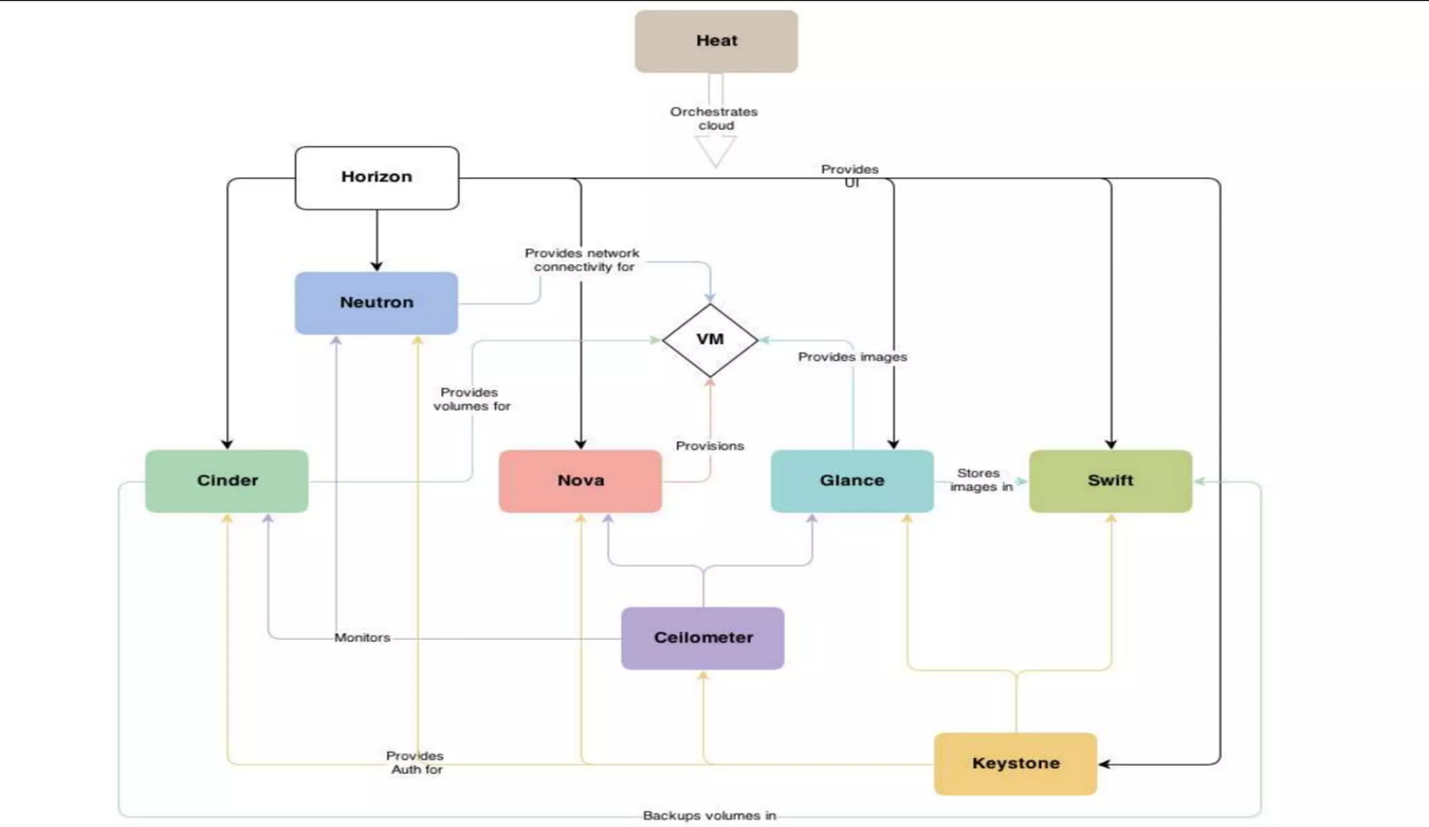
The components of OpenStack support each other by communicating through APIs. Each component is complementary, but not all components are required to create a basic cloud. For example, Nova (Compute) uses Glance (Image) to deploy virtual machines, Neutron (Networking) to provide connectivity, and Cinder (Block Storage) to attach volumes. Horizon (Dashboard) provides a web-based interface to manage all the other components. Ceilometer (Telemetry) collects measurements from various components and Heat (Orchestration) automates the creation of resource stacks.



*Image 4. Components interact with each other*

* **Core services:**

| **Service** | **Code** | **Description** |
| --- | --- | --- |
| Dashboard | Horizon | Web browser-based dashboard that you use to manage OpenStack services. |
| Identity | Keystone | Centralized service for authentication and authorization of OpenStack services and for managing users, projects, and roles. |
| Networking | Neutron | Provides connectivity between the interfaces of OpenStack services. |
| Block Storage | Cinder | Manages persistent block storage volumes for virtual machines. |
| Compute | Nova | Manages and provisions virtual machines running on hypervisor nodes. |
| Image | Glance | Registry service that you use to store resources such as virtual machine images and volume snapshots. |
| Object Storage | Swift | Allows users to store and retrieve files and arbitrary data. |
| Telemetry | Ceilometer | Provides measurements of cloud resources. |
| Orchestration | Heat | Template-based orchestration engine that supports automatic creation of resource stacks. |



*Image 5. Core services*

* 1. **Advantages and disadvantages**
* Advantages:

+ Cloud services enable collaboration

+ Few prerequisites are needed to use it: Cloud services can be used on any device with internet access.

+ Uniform standards

+ High data security and reliability

+ Scalable storage volume and performance

+ Access is not limited by location

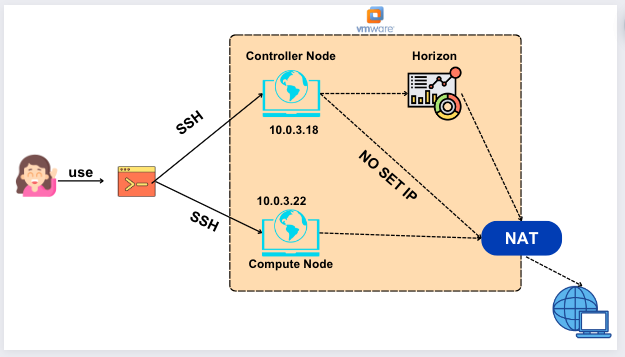
+ High distribution and acceptance

* Disadvantages:

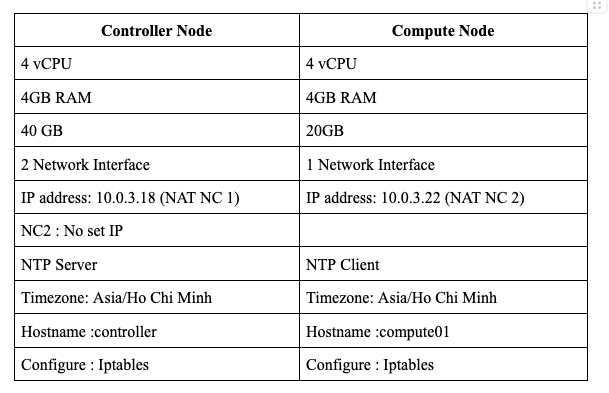
+Highly-dynamic range of functions: OpenStack is an open and very dynamic cloud-computing solution. New functions are added regularly, but some other functions can also be removed. OpenStack does not offer any sort of organized support.

+ OpenStack does not offer any sort of organized support. A lot of documentation and helpful articles can be found online, but it can take quite a bit of time and effort to find the right ones. Unlike commercial solutions, there is no dedicated customer support.

# **DEPLOYMENT MODEL AND CONDITIONS**



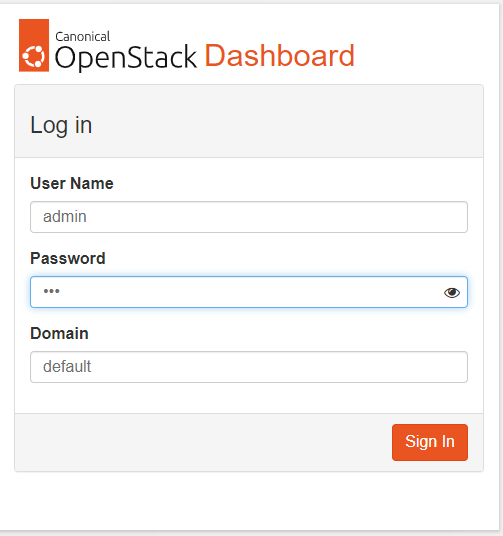
*Image 1. Deployment model*



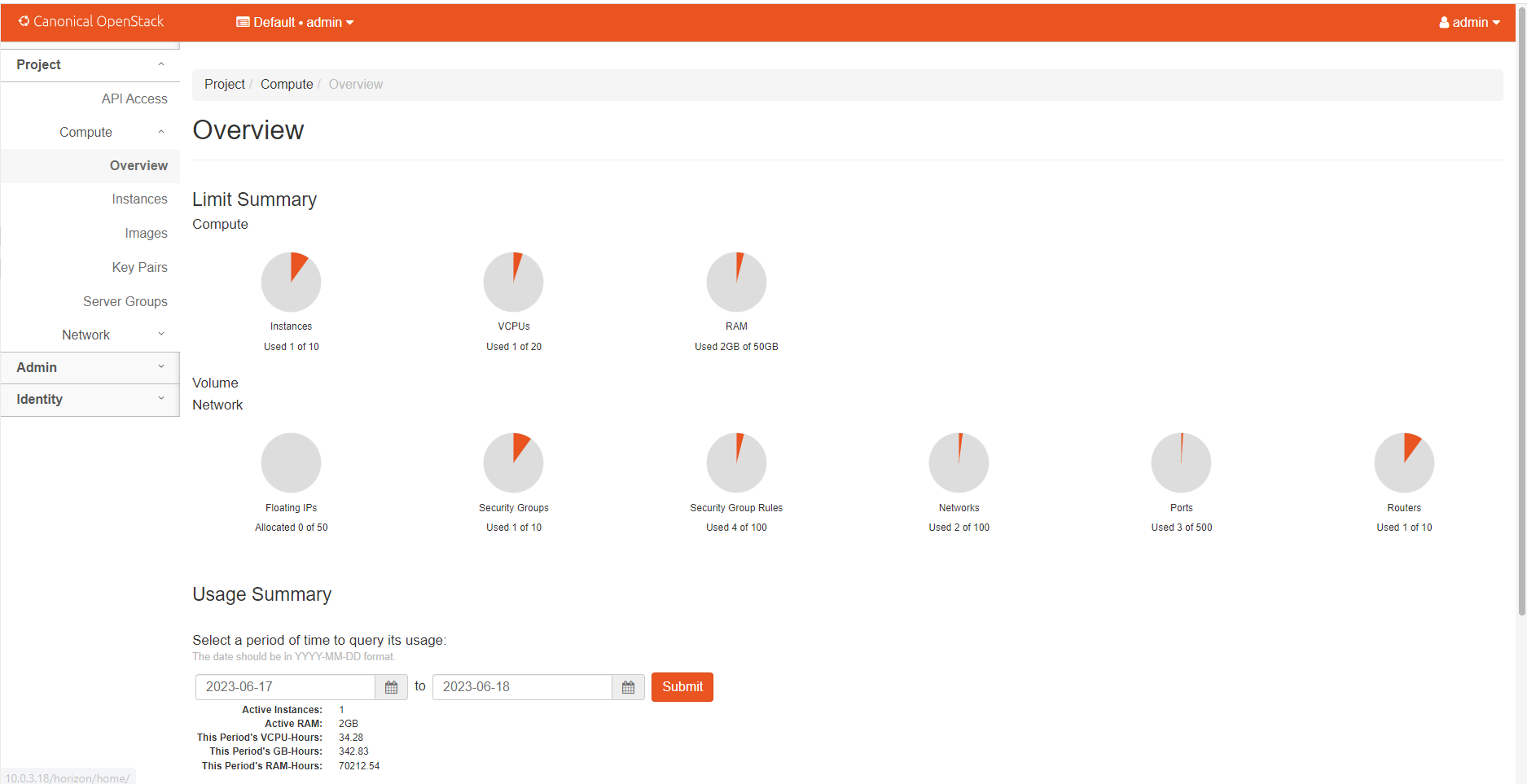
*Image 2. Set up environment*

# **CONCLUSION**

**3.1 – Result**



*Image 6. Login admin*

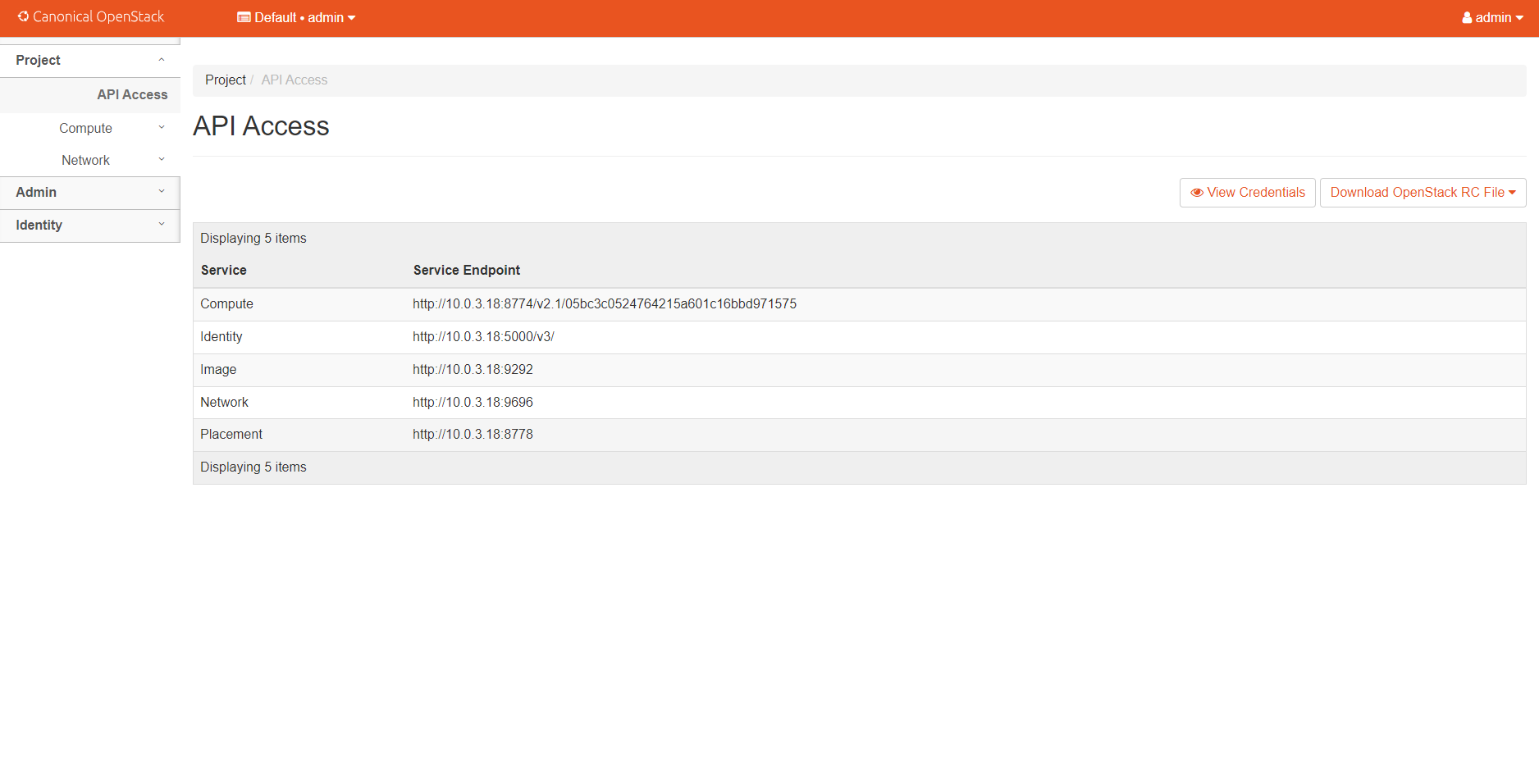


*Image 7. Admin interface*

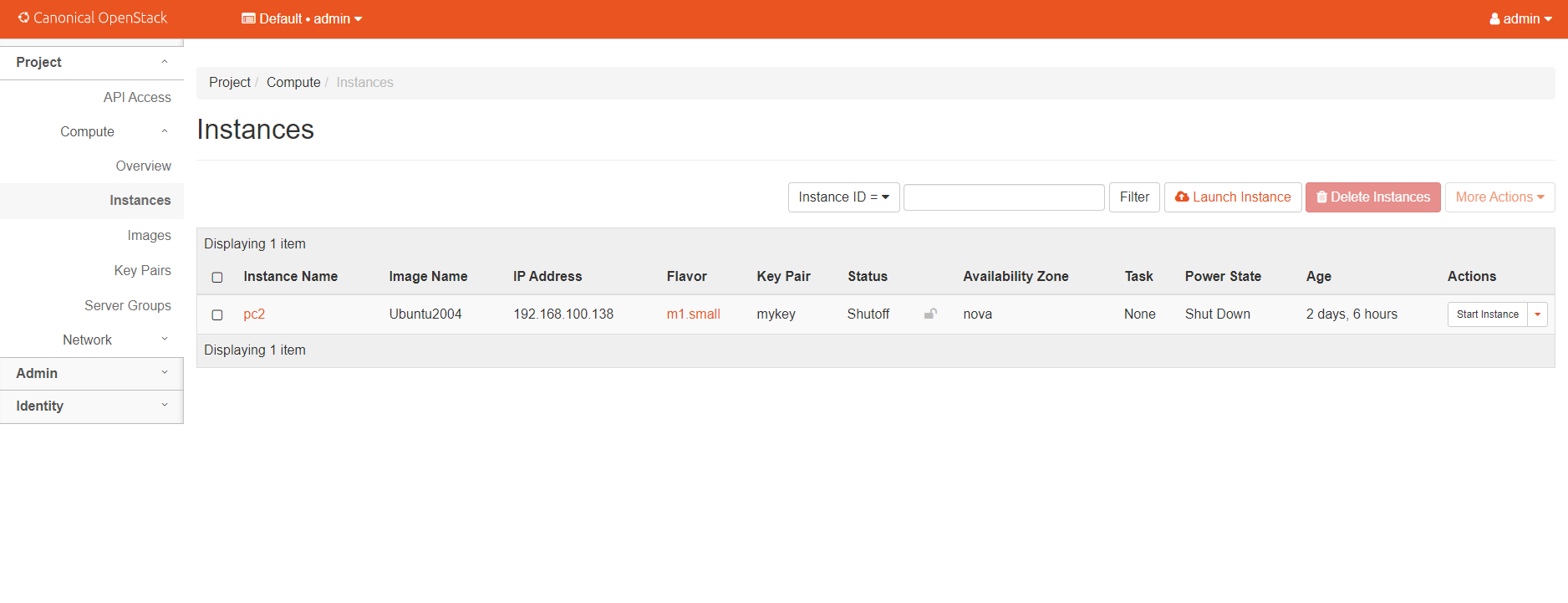
A screenshot of a computer

Description automatically generated with medium confidence

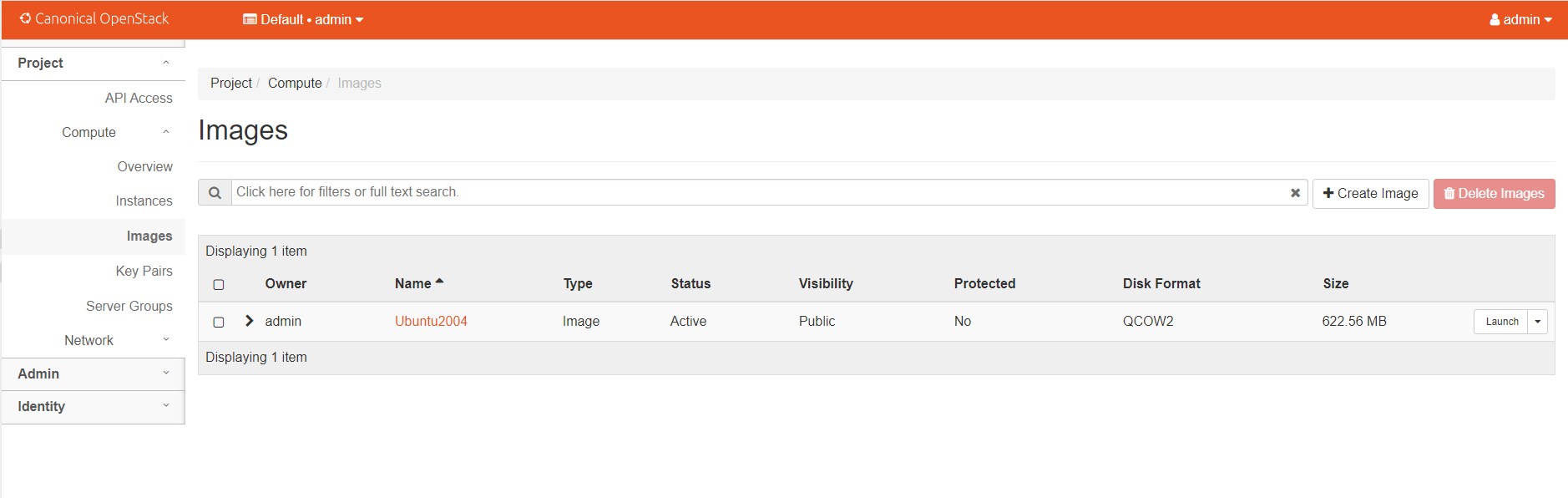
*Image 8. Usage statistics*



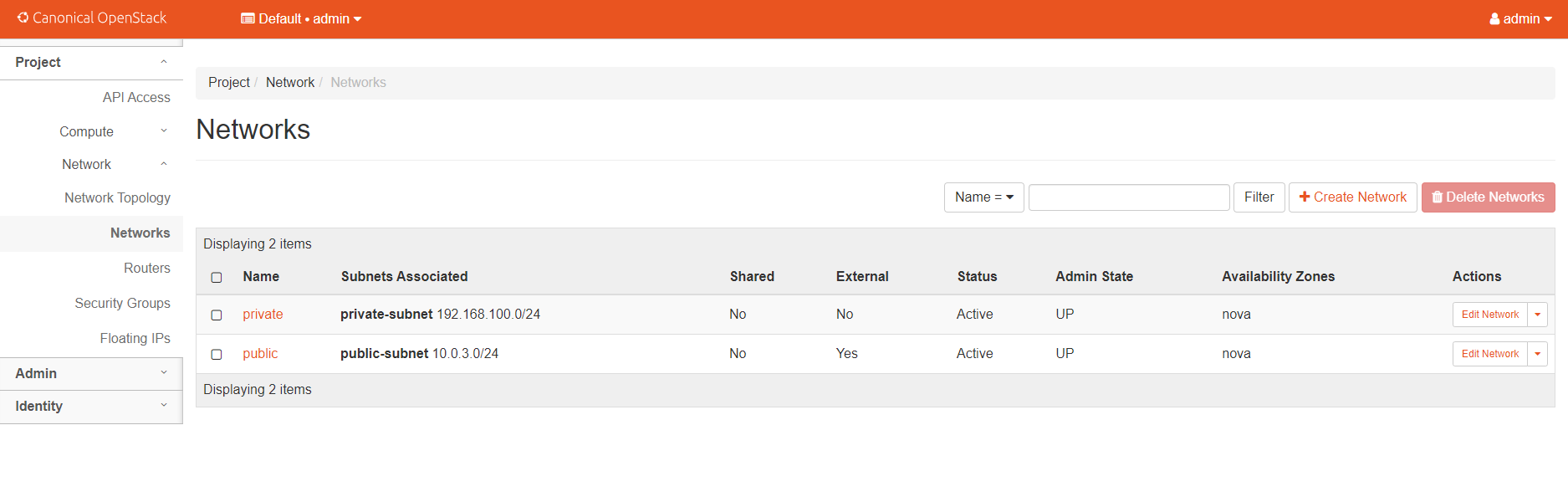
*Image 9. API Access*



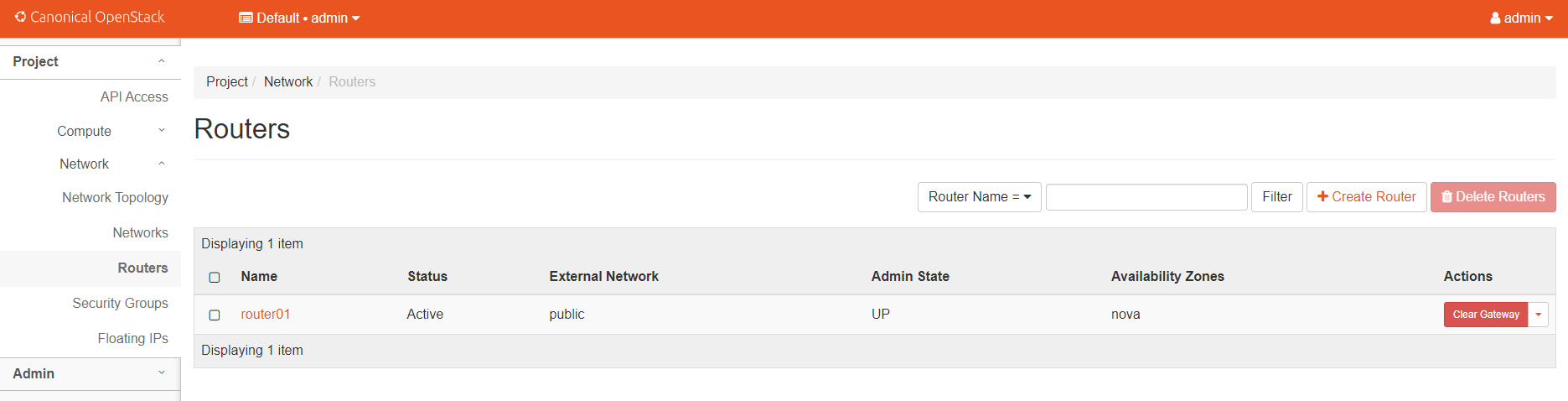
*Image 10. Instances statistics of Project*



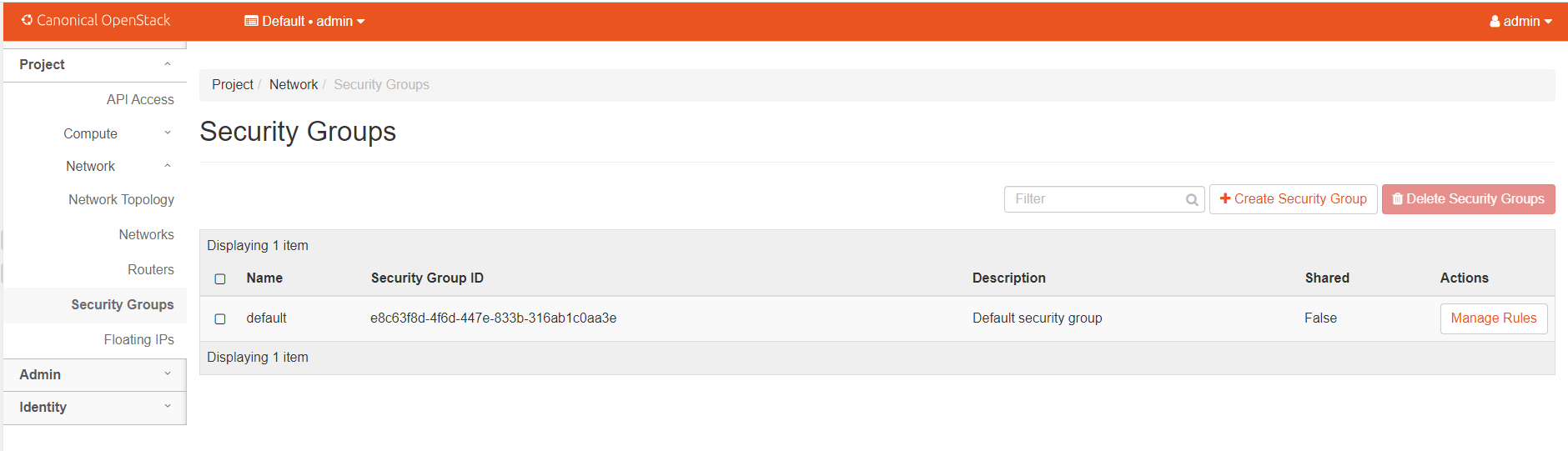
*Image 11. Images statistics of Project*



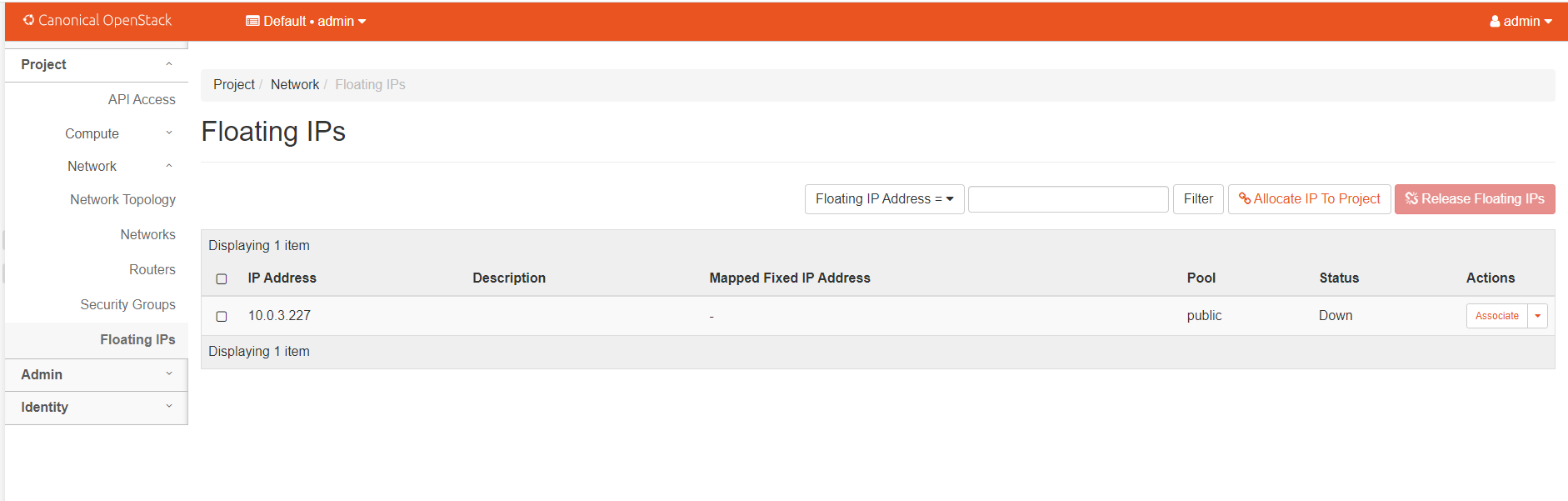
*Image 14. Networks statistics*



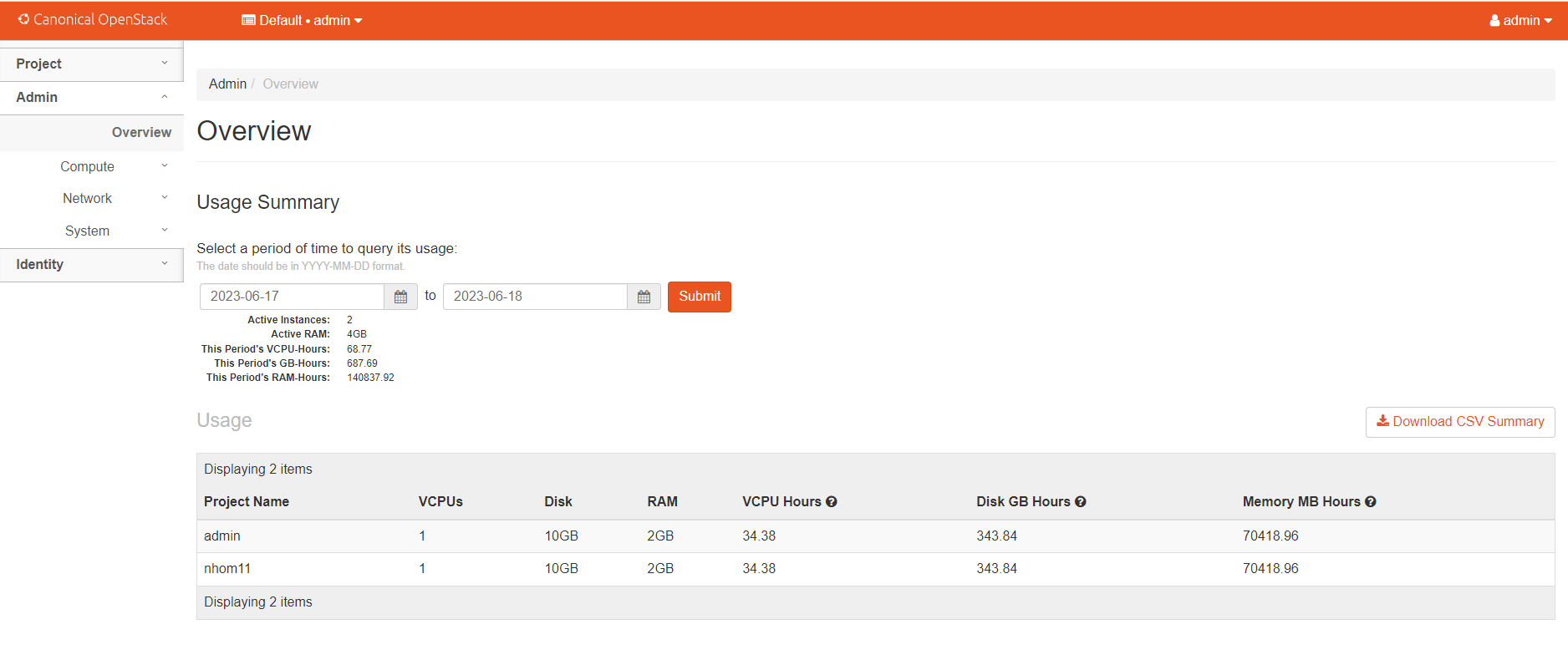
*Image 15. Routers statistics*



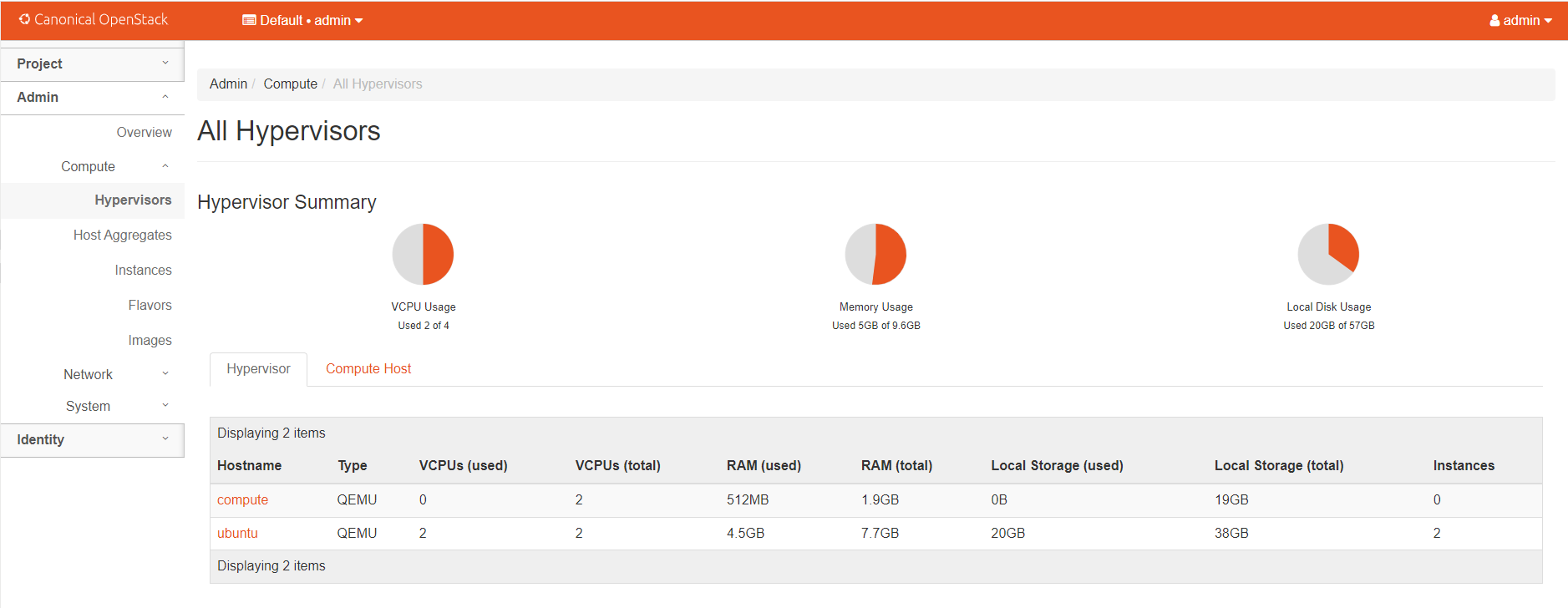
*Image 16. Security groups statistics*



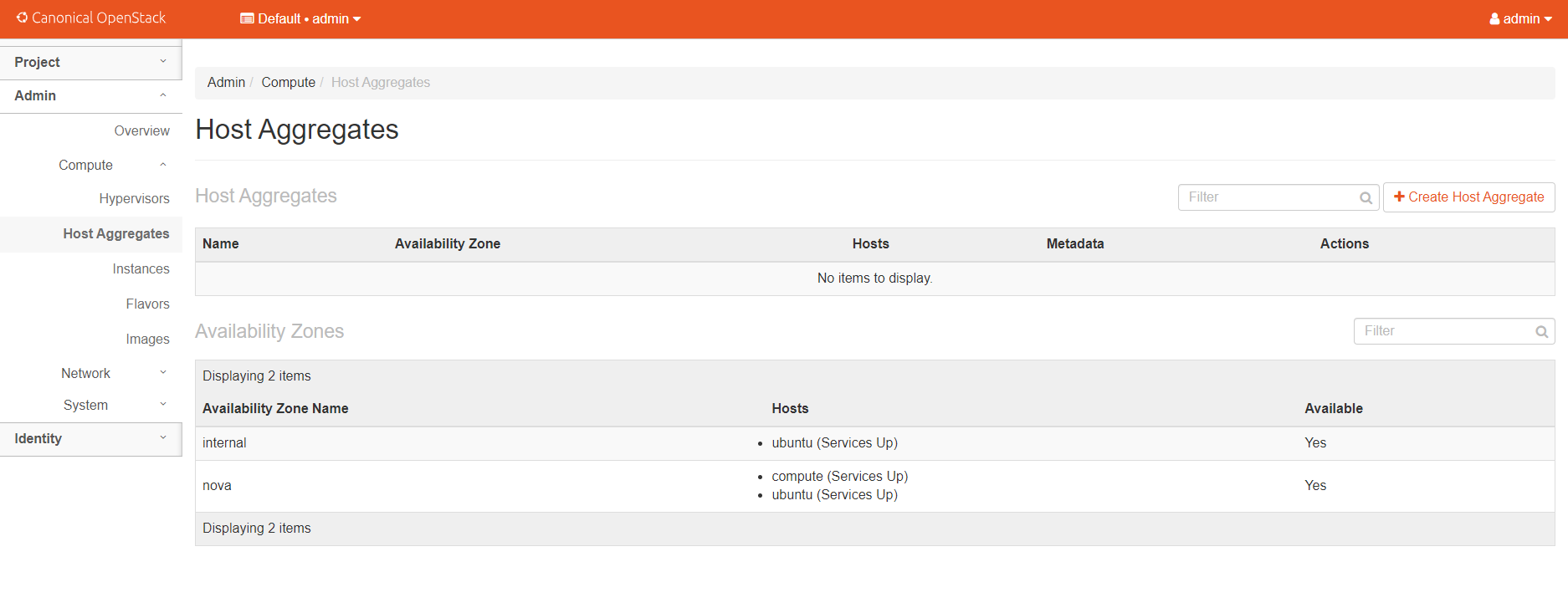
*Image 17. Floating IPs*



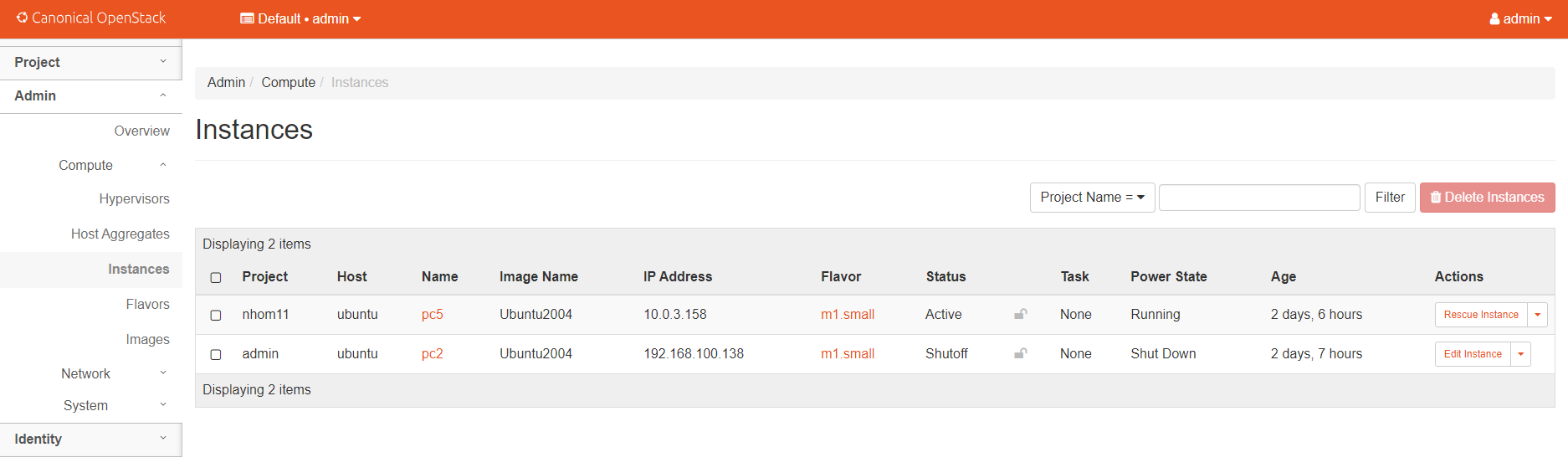
*Image 18. Overview*



*Image 19. All hypervisors*



*Image 20. Host aggregates*

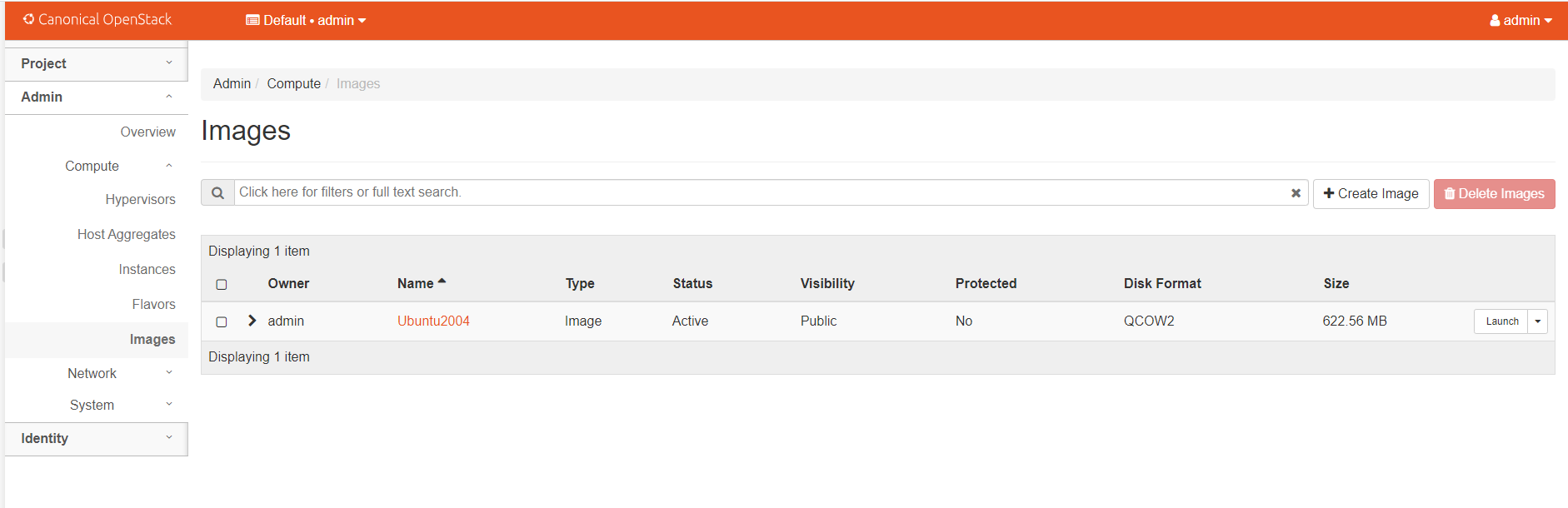


*Image 21. Instances statistics of Admin*

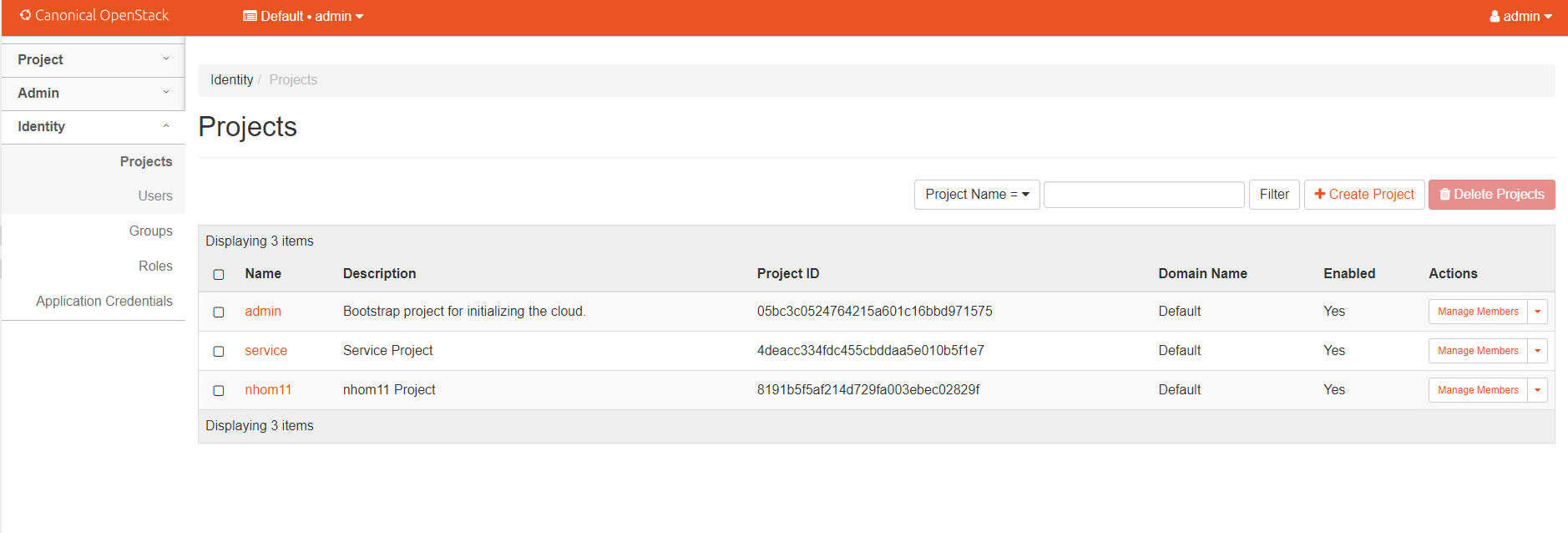
A screenshot of a computer

Description automatically generated with medium confidence

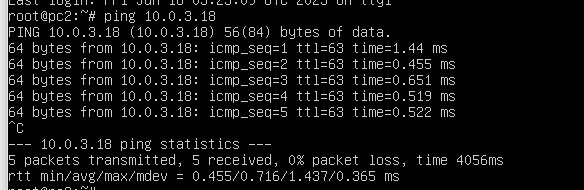
*Image 22. Images statistics of Admin*



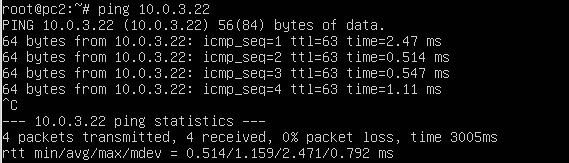
*Image 23. Flavors statistics of Admin*



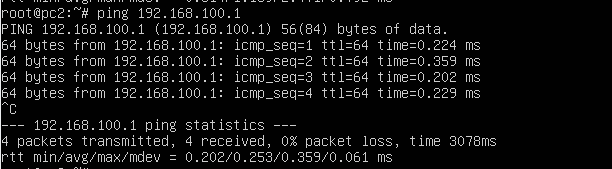
*Image 24. Identity of Project*



*Image 25. Ping from instance to controller node*



*Image 26. Ping from instance to compute node*



*Image 27. Ping to a location in network topology*

**DONE**