

# OpenStack security rules

Giuseppe Platania - INFN Catania  
Corso di formazione sulla Sicurezza Informatica  
Firenze 4-5/11/2015

# Outline

- Introduzione di Openstack
  - Breve storia
  - Componenti
  - Uenti
  - Tenant
  - Istanze
- Configurazione delle security roles
  - Security group
  - FWaaS

# Breve storia di Openstack

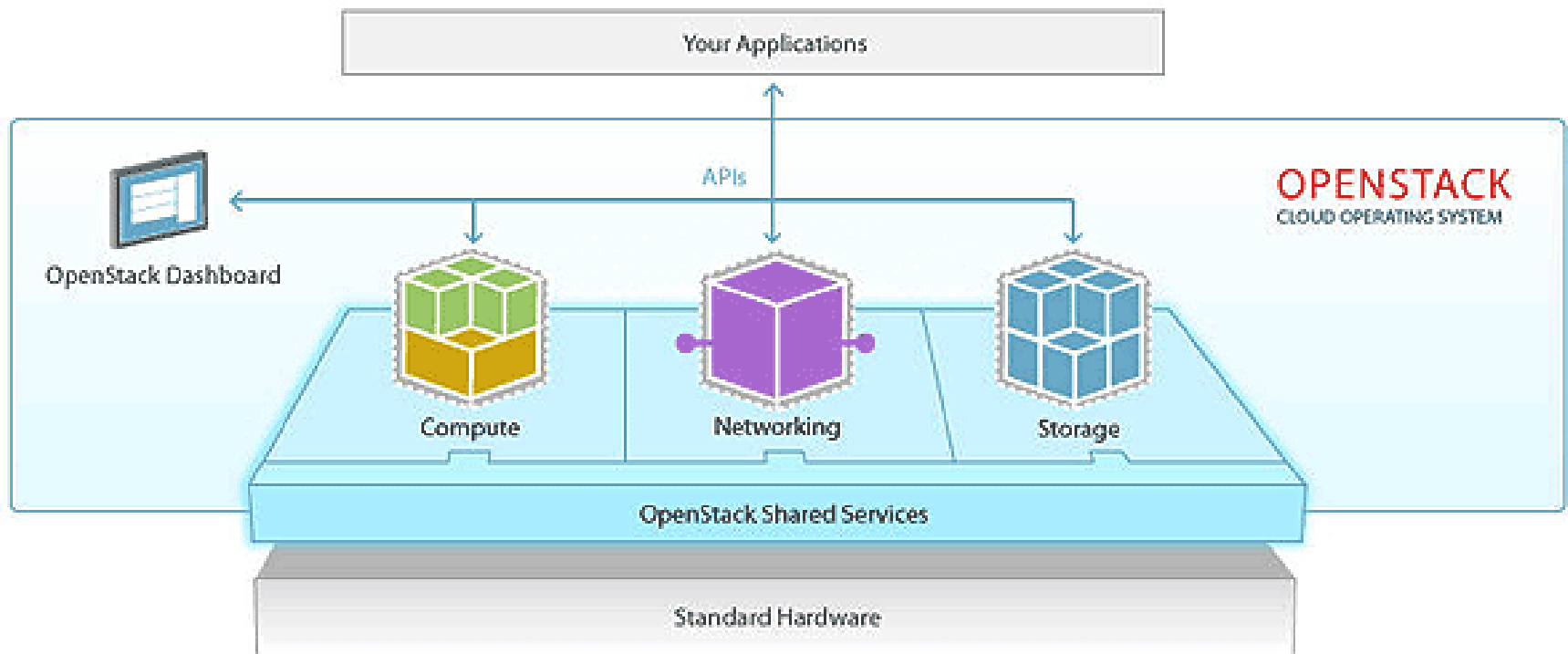
- è un **sistema operativo cloud**, modulare, in grado di offrire servizi di gestione di processi e storage secondo il modello **IaaS** (Infrastructure as a Service).
- È nato nel 2010 da una collaborazione tra NASA e Rackspace Cloud, che deve la sua rapida crescita a contributi provenienti da fronti diversi
- è totalmente **open source**, scritto in **Python** ed utilizza diversi altri software liberi; ciò ha contribuito a garantirgli il supporto di un'ampia comunità.
- Il suo sviluppo è dovuto al forte interesse di grandi realtà industriali come HP, Cisco, Dell, etc..
- Sono state rilasciate numerose *release*, l'ultima rilasciata è la **Liberty**
- Ogni sei mesi viene rilasciata una nuova versione ed organizzato un Summit internazionale

# Moduli di Openstack

L'architettura di Openstack è suddivisa nei seguenti moduli principali:

- Compute (Nova)
- Identity (Keystone)
- Image Service (Glance)
- Networking (Neutron)
- Dashboard (Horizon)
- Block Storage (Cinder)

# Principali componenti



# NOVA (Compute)

- si occupa di far funzionare e gestire le VM (chiamate **istanze**)
- supporta diversi tool di virtualizzazione
  - libvirtd, qemu, vSphere, lxc, docker, Hyper-V, Xen
  - **libvirtd** (KVM) è la più utilizzata.
- interagisce con gli altri moduli e si preoccupa di trovare (scheduling) un hypervisor adatto, mette le attività in coda (queueing), alloca le risorse, istanzia l'immagine, aggancia le reti, associa i volumi, etc..
- gestisce gli errori
- configurazione centralizzata della sicurezza (Security rules).

# NOVA (Compute)

- Le operazioni tipiche di cui si occupa Nova sono:
  - Lanciare istanze
  - Terminare istanze
  - Reboot e stop
  - Attach/detach di volumi
  - applica security groups
  - Console
  - Log
- Si può interagire con Nova attraverso **Horizon** o con la **CLI**.

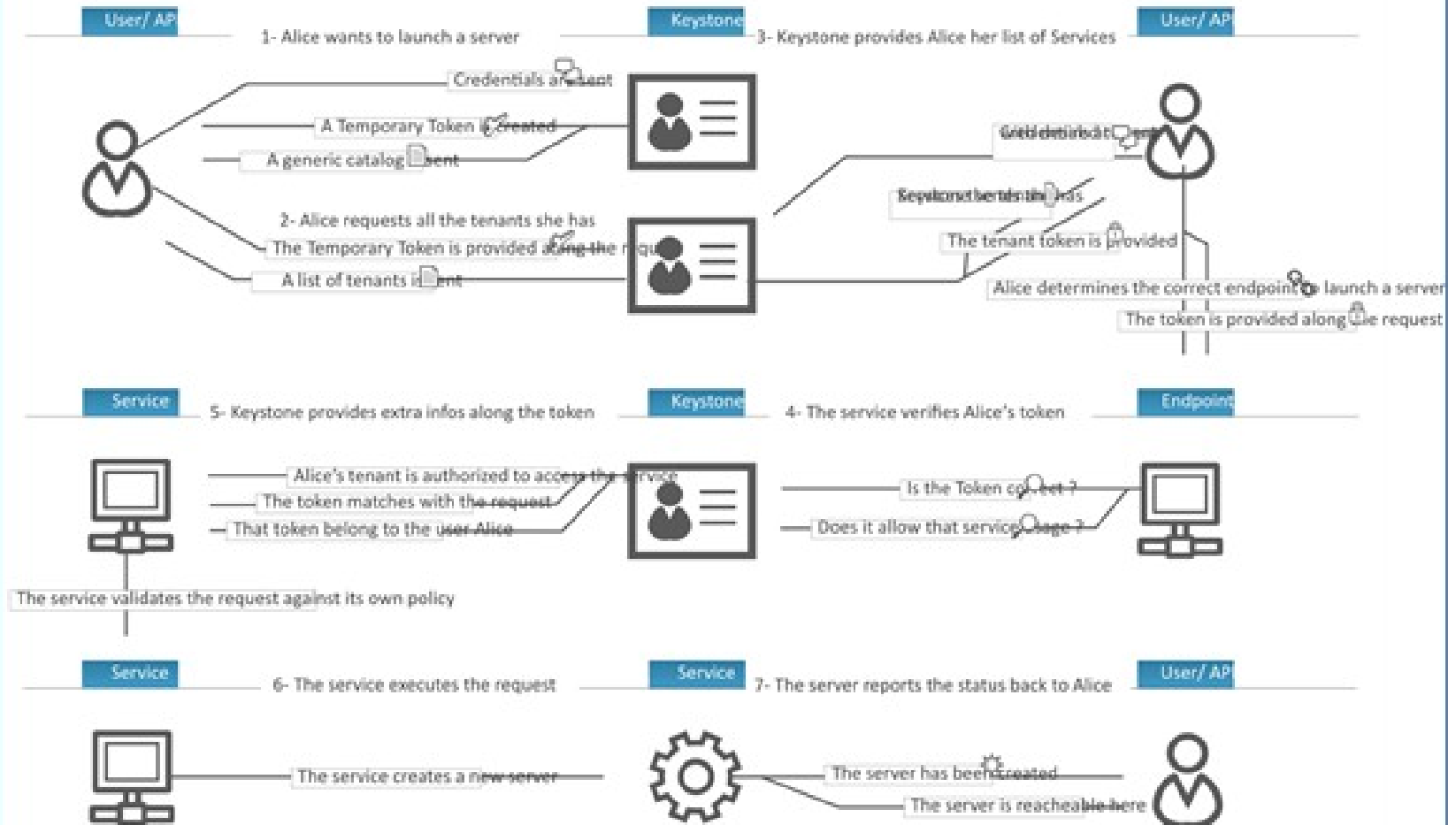
# KEYSTONE (Identity)

- si occupa di gestire i tenant, utenti, gruppi, ruoli, autorizzazioni, autenticazione
- Registra gli endpoint dei servizi con cui si interfaccia ad ogni richiesta dell'utente.
- ogni volta che si effettua una qualsiasi operazione su un qualsiasi servizio (come gestire le VM, lo storage, la rete), Keystone genera e gestisce i **token** delle API, verifica i permessi e permette o impedisce che le operazioni vadano a buon fine.



# KEYSTONE (Identity)

The Keystone Identity Manager



# Utenti

- l'utente su openstack è la rappresentazione digitale della persona che utilizza i servizi e può essere suddiviso in:
  - amministratore dell'iaaS (admin)
  - amministratore del tenant (member)
- Per ogni utente viene definito il ruolo che determina le policy di accesso a cosa e l'utilizzo di quali risorse

# TENANT

- Il Tenant è un progetto definito dall'amministratore della cloud al quale vengono assegnate le risorse (CPU, Storage, RAM) che l'utente avrà a disposizione.
- Openstack è un IaaS multi-tenant
- Il tenant e utente admin hanno il controllo dell'intera CLOUD
- più utenti possono accedere al tenant

# Role

- set di privilegi assegnati all'utente per compiere determinate operazioni
- quando keystone genera il token per l'utente, esso include una lista di role che servono ai servizi (nova, neutron, etc.) a determinare cosa è autorizzato a fare
- di default esistono admin che ha ruoli per amministrare la IaaS, e member per amministrare il tenant assegnato all'utente

# token

- viene generato da keystone ad ogni richiesta di accesso ai servizi da parte dell'utente
- il servizio “legge” il token e controlla se è autorizzato o no a svolgere l'operazione richiesta
- ha una validità finita e può essere revocato

# NEUTRON (Network)

- è il modulo che si occupa delle configurazioni di rete.
- È nato recentemente, con il nome di *Quantum*, a partire dalla versione Folsom. In precedenza, veniva usato *nova-network*, servizio di Nova, ma ciò comportava dei limiti, specialmente per configurazioni avanzate. Pertanto è stato isolato in un modulo che si occupa dell'infrastruttura di comunicazione sia tra componenti interni che verso l'esterno
- L'utente amministratore del tenant può creare topologie di rete ed agganciarli ad una rete esterna tramite un router virtuale
- usa openvswitch con ml2 plugin
- supporta VLAN e IPv6
- load balancer, firewall, vpn

# GLANCE (Image Service)

- gestisce il database delle immagini virtuali che vengono utilizzate poi da nova per creare e lanciare macchine virtuali
- utilizza solitamente un database come storage
- supporta diversi tipi di immagini:
  - ami, ari, aki, vhd, vmdk, raw, qcow2, vdi, iso

# CINDER (Block Storage)

- è il modulo che gestisce i volumi a blocchi per il salvataggio dei dati. Nelle prime release era integrato in nova (*nova-volume*)
- è configurabile dalla dashboard
- permette la creazione di volumi di storage che possono essere “agganciati” alle macchine virtuali
- è possibile creare volumi di boot per lanciare le macchine virtuali
- si interfaccia con numerose piattaforme di storage quali IBM Storwize, GlusterFS, etc..
- è possibile creare lo snapshot del volume



# HORIZON (Dashboard)

- è l'interfaccia Web con cui dialoga l'utente.
- È realizzata mediante web Django che tipicamente viene messa in produzione con Apache WSGI
- permette di gestire sia dal punto di vista dell'amministratore che degli utenti, i progetti (tenant), lavorare con le istanze (VM), i volumi, le immagini, gli snapshot, la rete, le security rules, etc..

# HORIZON (Dashboard)






openstack Group1 gr1 Sign Out

Project

- Compute
- Overview
- Instances
- Images
- Access & Security
- Network
- Identity

## Overview

### Limit Summary

 <p><b>Instances</b> Used 5 of 10</p>	 <p><b>VCPUs</b> Used 6 of 20</p>	 <p><b>RAM</b> Used 6GB of 50GB</p>	 <p><b>Floating IPs</b> Used 0 of 50</p>	 <p><b>Security Groups</b> Used 1 of 10</p>
--	--	--	---	--

### Usage Summary

Select a period of time to query its usage:

From:  To:   The date should be in YYYY-mm-dd format.

**Active Instances: 5 Active RAM: 6GB This Period's VCPU-Hours: 834.31 This Period's GB-Hours: 5066.17**

### Usage

[Download CSV Summary](#)

Instance Name	VCPUs	Disk	RAM	Time since created
<a href="#">Pippo</a>	1	1GB	512MB	1 week
<a href="#">Ciccio</a>	1	1GB	512MB	1 week
<a href="#">cmsrt</a>	1	1GB	512MB	1 week
<a href="#">PLT</a>	1	1GB	512MB	1 week
<a href="#">SL6x-x86_64</a>	2	40GB	4GB	4 days, 12 hours

Displaying 5 items

# HORIZON (Dashboard)

The screenshot displays the OpenStack Horizon dashboard interface. At the top left is the OpenStack logo. The top navigation bar includes a 'Group1' dropdown menu, a user profile dropdown for 'gr1', and a 'Sign Out' button. A left-hand sidebar contains a 'Project' dropdown menu with sub-items: 'Compute', 'Overview', 'Instances', 'Images' (highlighted with a red bar), 'Access & Security', 'Network', and 'Identity'. The main content area is titled 'Images' and features a filter bar with 'Project (0)', 'Shared with Me (0)', and 'Public (2)' filters, along with '+ Create Image' and 'Delete Images' buttons. Below the filter bar is a table with two columns: 'Image Name' and 'Actions'. The table contains two rows of image data.

<input type="checkbox"/>	Image Name	Type	Status	Public	Protected	Format	Size	Actions
<input type="checkbox"/>	<a href="#">sl6x-x86_64-boot</a>	Image	Active	Yes	No	QCOW2	231.0 MB	<input type="button" value="Launch"/>
<input type="checkbox"/>	<a href="#">cirros-0.3.3-x86_64</a>	Image	Active	Yes	No	QCOW2	12.6 MB	<input type="button" value="Launch"/>

Displaying 2 items

# HORIZON (Dashboard)

openstack Group1 gr1 Sign Out

Project Compute Overview Instances Images Access & Security Network Identity

## Instances

Instances Instance Name Filter Filter Launch Instance Soft Reboot Instances Terminate Instances

<input type="checkbox"/>	Instance Name	Image Name	IP Address	Size	Key Pair	Status	Availability Zone	Task	Power State	Time since created	Actions
<input type="checkbox"/>	SL6x-x86_64	sl6x-x86_64-boot	172.20.1.13	m1.medium	-	Active	nova	None	Running	4 days, 12 hours	Create Snapshot
<input type="checkbox"/>	PLT	cirros-0.3.3-x86_64	10.20.1.12	m1.tiny	controller-key	Shutoff	nova	None	Shut Down	1 week	Start Instance
<input type="checkbox"/>	cmsrt	cirros-0.3.3-x86_64	172.20.1.12	m1.tiny	network-key	Shutoff	nova	None	Shut Down	1 week	Start Instance
<input type="checkbox"/>	Ciccio	cirros-0.3.3-x86_64	10.20.1.10	m1.tiny	-	Shutoff	nova	None	Shut Down	1 week	Start Instance
<input type="checkbox"/>	Pippo	cirros-0.3.3-x86_64	172.20.1.10	m1.tiny	-	Shutoff	nova	None	Shut Down	1 week	Start Instance

Displaying 5 items

# HORIZON (Dashboard)

## Launch Instance

Details \* Access & Security \* Networking \* Post-Creation

Advanced Options

## Launch Instance

Details \* Access & Security \* **Networking \*** Post-Creation

Advanced Options

**Selected networks**

**Available networks**

- gr1-net3 (ad3cf087-ae16-4d39-a715-e12072f1dce2) +
- gr1-net (1c9aa338-7384-4746-a645-1059d9804579) +
- gr1-net2 (4ae4fb5f-4c1d-45db-a59c-509fa5ddd740) +

Choose network from Available networks to Selected networks by push button or drag and drop, you may change NIC order by drag and drop as well.

Cancel Launch

# HORIZON (Dashboard)

openstack Group1 gr1 Sign Out

Project

- Compute
- Network

Network Topology

Small Normal

Launch Instance Create Network Create Router

The diagram illustrates a network topology with four vertical bars representing networks: ext-net (blue), gr1-net3 (orange), gr1-net (green), and gr1-net2 (red). ext-net is connected to gr1-net3 via a router icon. gr1-net3 is connected to gr1-net via a router icon. gr1-net is connected to gr1-net2 via a router icon. Each network has associated IP addresses and icons representing instances or services.

- ext-net
- gr1-net3: 10.10.10.0/24
- gr1-net: 177.20.1.0/24
- gr1-net2: 10.20.1.0/24

# Tipica installazione

- Controller node
  - Keystone
  - Nova
  - Glance
  - Horizon
- Network node
  - Neutron
- Compute node
  - Nova

# Istanze

- rappresenta la macchina virtuale in esecuzioni sull'hypervisor
- viene lanciata da NOVA richiamando un'immagine tramite GLANCE
- è necessario specificare un FLAVOR.
- FLAVOR= profilo che definisce le caratteristiche della macchina virtuale in termini di:
  - numero di CPU
  - dimensione RAM e disco
- è possibile aggiungere/rimuovere spazio disco e floating IP
- applicazione di security group



# Security Groups

- sono un insieme di regole ip filter che vengono applicate sull'istanza in fase di creazione della stessa e possono essere modificate successivamente dall'utente amministratore del tenant
- se non viene specificato, di default viene applicato un security group che blocca il traffico
- i membri del tenant possono creare e modificare questi gruppi

# Security Groups

The screenshot shows the OpenStack dashboard interface. At the top left is the OpenStack logo. The top right shows the user 'gr1' and a 'Sign Out' button. A left-hand navigation menu includes 'Project', 'Compute', 'Overview', 'Instances', 'Images', 'Access & Security' (highlighted), 'Network', and 'Identity'. The main content area is titled 'Manage Security Group Rules: test' and 'Security Group Rules'. It features two buttons: '+ Add Rule' and 'X Delete Rules'. Below these is a table with two rows of rules. The table has columns for checkboxes, Direction, Ether Type, IP Protocol, Port Range, Remote, and Actions. The first row shows an Egress rule for IPv6 with remote address ::/0 (CIDR). The second row shows an Egress rule for IPv4 with remote address 0.0.0.0/0 (CIDR). Below the table, it says 'Displaying 2 items'.

Project ▼

Group1 ▼ gr1 ▼ Sign Out

Compute ▼

Overview

Instances

Images

Access & Security

Network ▶

Identity ▶

## Manage Security Group Rules: test

### Security Group Rules

[+ Add Rule](#) [X Delete Rules](#)

<input type="checkbox"/>	Direction	Ether Type	IP Protocol	Port Range	Remote	Actions
<input type="checkbox"/>	Egress	IPv6	Any	-	::/0 (CIDR)	<a href="#">Delete Rule</a>
<input type="checkbox"/>	Egress	IPv4	Any	-	0.0.0.0/0 (CIDR)	<a href="#">Delete Rule</a>

Displaying 2 items

# Security Groups

## Add Rule

### Rule \*

Custom TCP Rule

### Direction

Ingress

### Open Port \*

Port

### Port ?

### Remote \* ?

CIDR

### CIDR ?

0.0.0.0/0

## Add Rule

### Rule \*

SSH

### Remote \* ?

CIDR

### CIDR ?

0.0.0.0/0

## Add Rule

### Rule \*

SSH

### Remote \* ?

Security Group

### Security Group

default

### Ether Type

IPv4

## Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

**Rule:** You can specify the desired rule template or use custom rules; the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

**Open Port/Port Range:** For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending ports for the range. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

**Remote:** You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

Cancel

Add

Cancel

Add

# Security Groups

openstack Group1 gr1 Sign Out

Project

- Compute
- Overview
- Instances
- Images
- Access & Security
- Network
- Identity

## Manage Security Group Rules: test

Security Group Rules + Add Rule ✕ Delete Rules

<input type="checkbox"/>	Direction	Ether Type	IP Protocol	Port Range	Remote	Actions
<input type="checkbox"/>	Egress	IPv6	Any	-	:::0 (CIDR)	Delete Rule
<input type="checkbox"/>	Egress	IPv4	Any	-	0.0.0.0/0 (CIDR)	Delete Rule
<input type="checkbox"/>	Ingress	IPv4	TCP	22 (SSH)	10.20.1.0/24 (CIDR)	Delete Rule

Displaying 3 items

# Security Groups

## Launch Instance ×

Details \*

Access & Security \*

Networking \*

Post-Creation

Advanced Options

### Key Pair ?

Select a key pair

+

Control access to your instance via key pairs, security groups, and other mechanisms.

### Security Groups \* ?

default

test

Cancel

Launch



# Laboratorio

- Dividersi in 10 gruppi da 4
- ogni gruppo avrà a disposizione:
  - un Tenant
  - un utente amministratore del Tenant
  - infrastruttura di rete con due subnet agganciate alla external network tramite router virtuale
  - 1 VM con floating IP pubblico e IP privato su una delle due subnet per accedere alle macchine CE-n e WN-n per i test sui security groups
- credenziali uniche:
  - username gr(n), password WSgr(n)

# Laboratorio

- Accesso alla dashboard di openstack
  - <http://193.206.208.84/dashboard>
- Bastion host
  - Group 1 ---> 193.206.219.244
  - Group 2 ---> 193.206.219.243
  - Group 3 ---> 193.206.219.242
  - Group 4 ---> 193.206.219.241
  - Group 5 ---> 193.206.219.240
  - Group 6 ---> 193.206.219.239
  - Group 7 ---> 193.206.219.238
  - Group 8 ---> 193.206.219.232
  - Group 9 ---> 193.206.219.236
  - Group 10 ---> 193.206.219.237

# Accesso al bastion host e primi test

- ssh gr(n)@193.206.219.xxx
  - ping 172.20.(n).xxx
  - telnet 172.20.(n).xxx 22
  - ping 10.20.(n).xxx
  - telnet 10.20.(n).xxx 22



# Test del server pbs

- su - ops001
- vi hostname.sh

```
#!/bin/sh
```

```
sleep 5
```

```
hostname
```

- qsub -q cert hostname.sh
- qstat -an
- log
  - /var/spool/pbs/server\_logs/20151105

# Porte da abilitare su CE e WN

<https://twiki.cern.ch/twiki/pub/EMI/GenericInstallationConfigurationEMI3/middle-ware-ports.txt>

- CE
  - ingresso
    - SSH → tcp 22
    - PBS server → tcp 15001
  - uscita
    - any verso WN
- WN
  - ingresso
    - SSH → tcp 22
    - PBS mom → tcp 15002
  - uscita
    - any verso CE

# FWaaS

- è un plugin di neutron che aggiunge un firewall perimetrale
- utilizza iptables per applicare le policy ai router virtuali
- si differisce dai security groups in quanto non agisce direttamente sulle istanze
- <https://wiki.openstack.org/wiki/Neutron/FWaaS/HowToInstall>