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

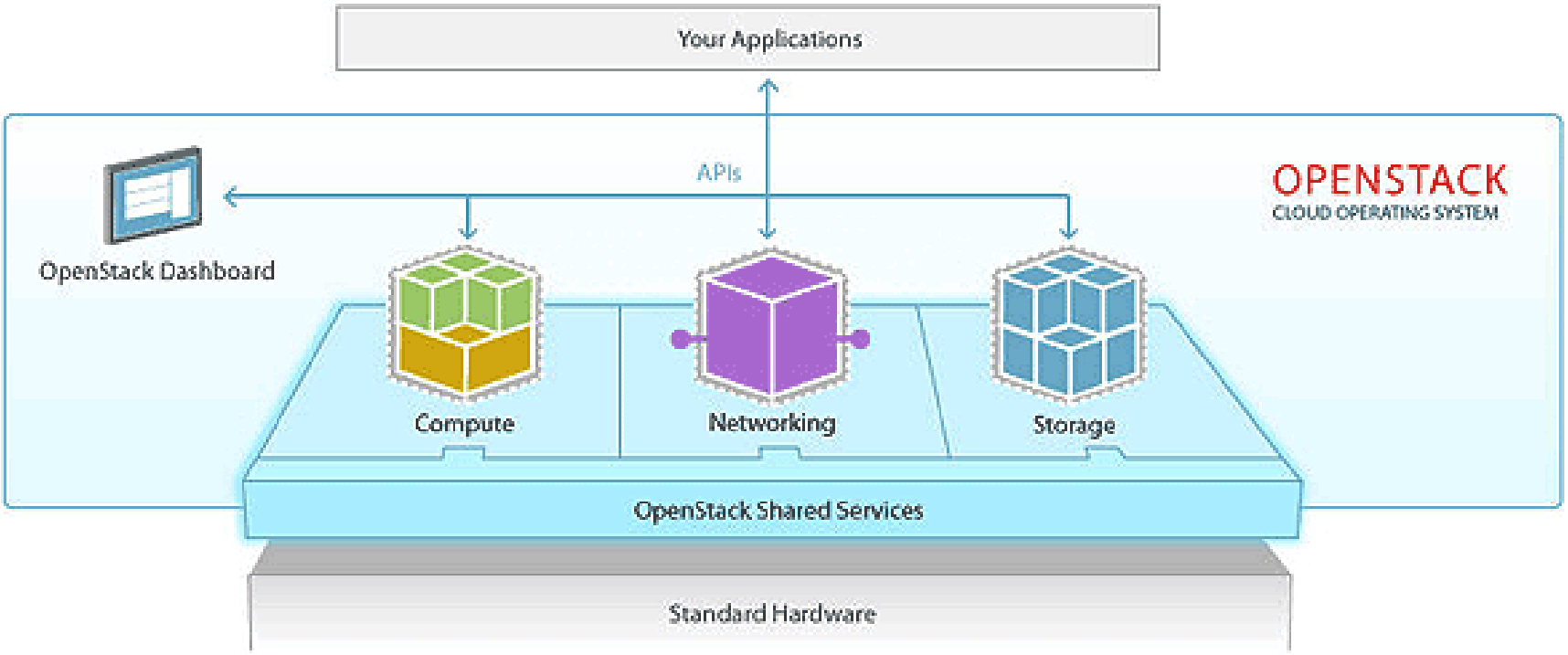
- è un **sistema operativo cloud**, modulare, in grado di offrire servizi di gestione di processi e storage secondo il modello **IaaS** (**I**nfrastructure **a**s **a** **S**ervice).
- È nato nel 2010 da una collaborazione tra NASA e Rackspace Cloud, che deve la sua rapida crescita a contributi provenienti 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).

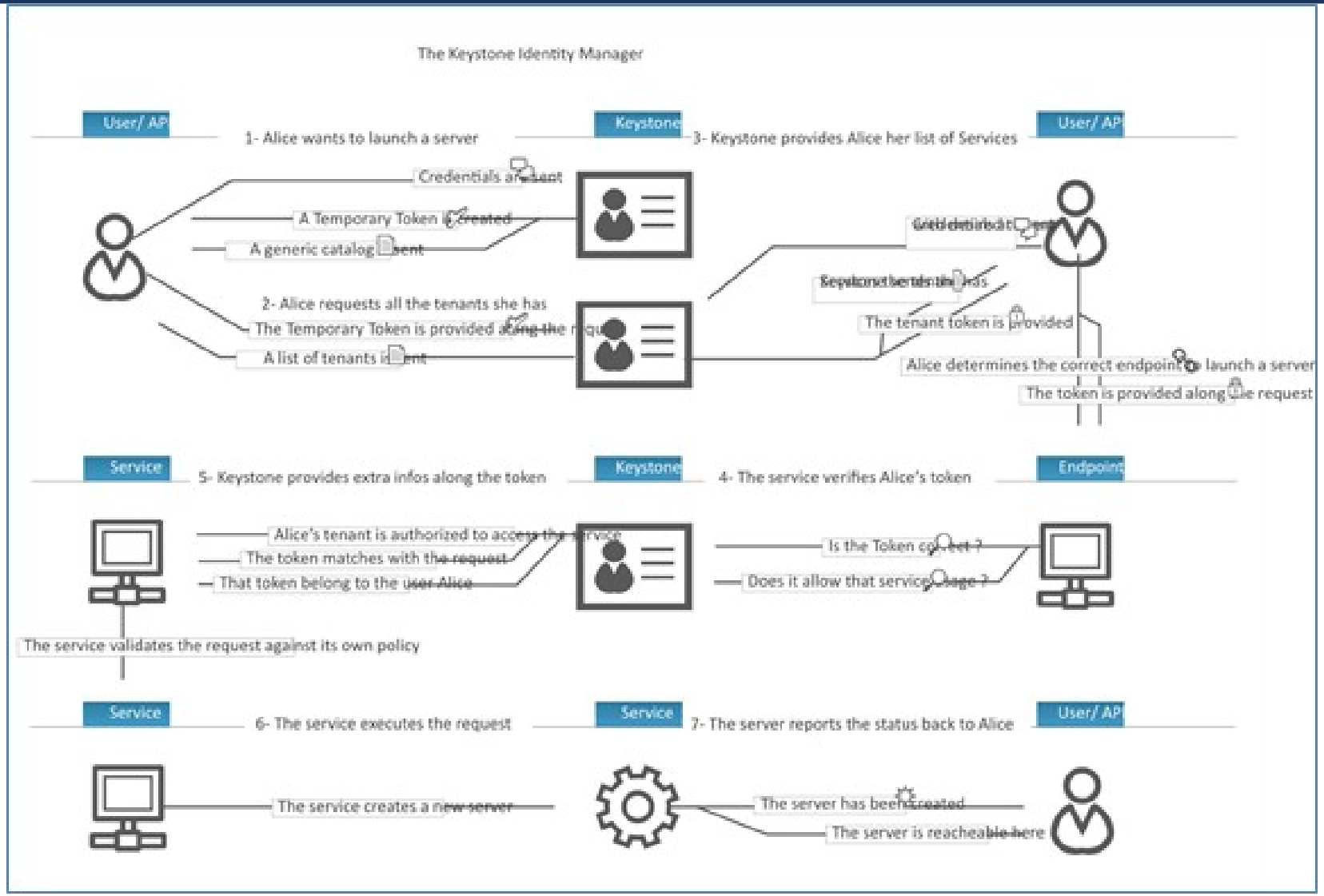
NOVA (Compute)

- 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)



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

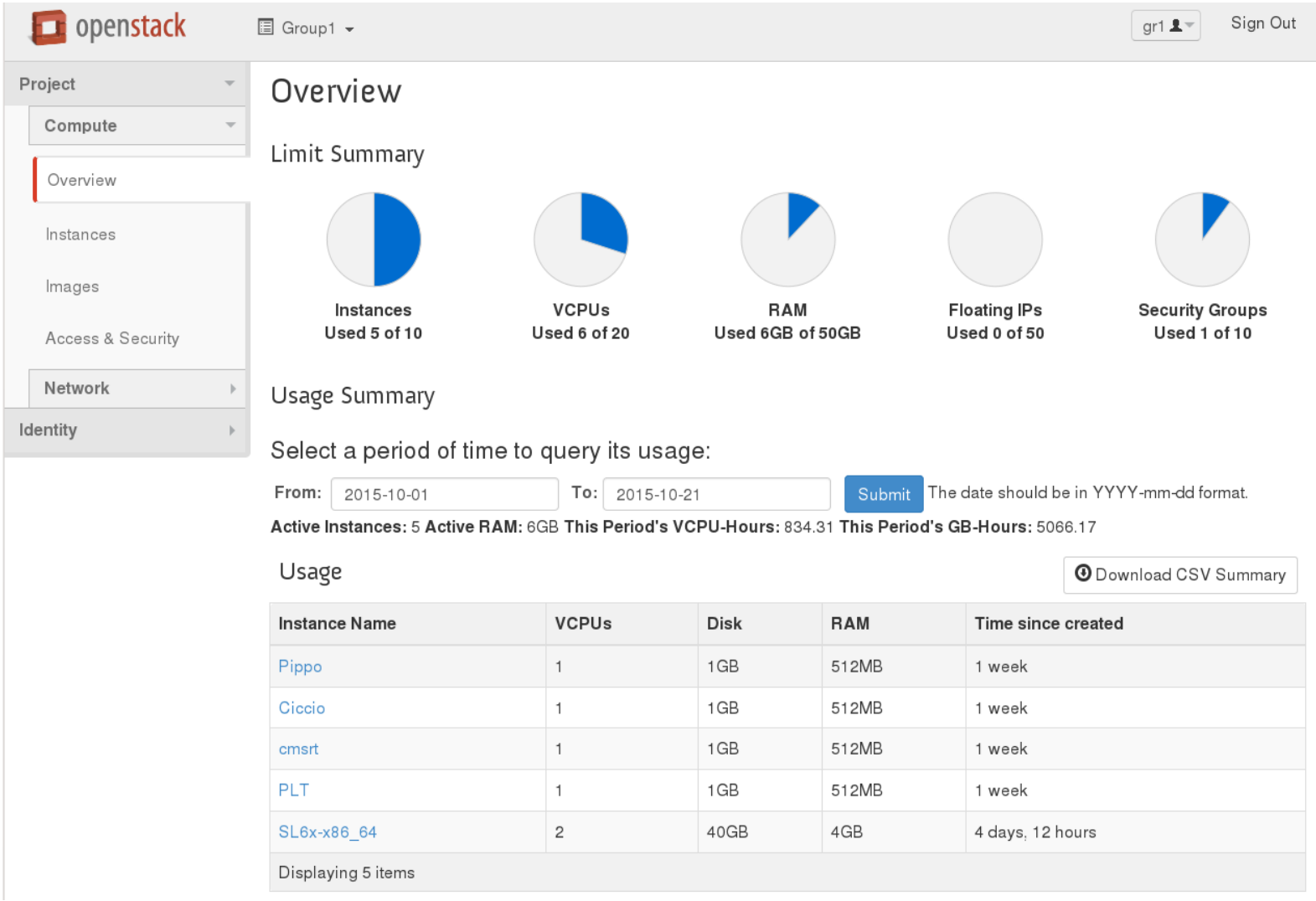
CINDER (Block Storage)

- 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

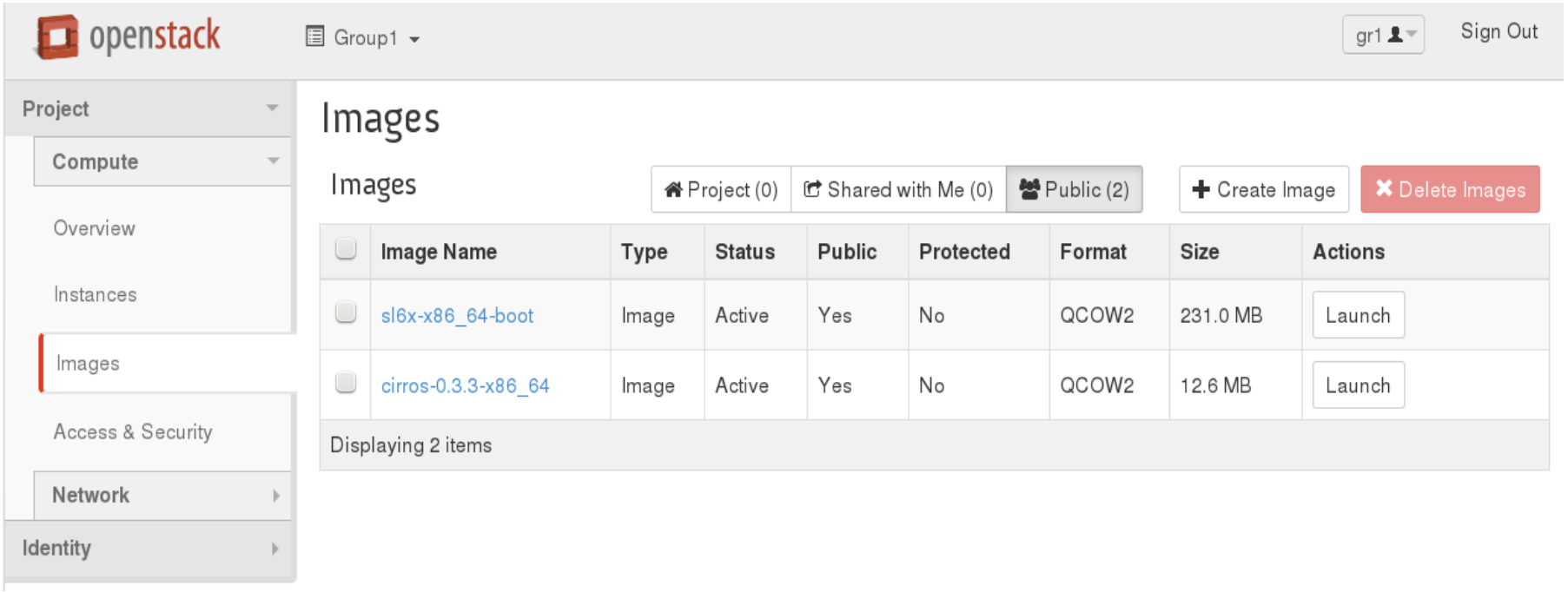
HORIZON (Dashboard)

- 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..

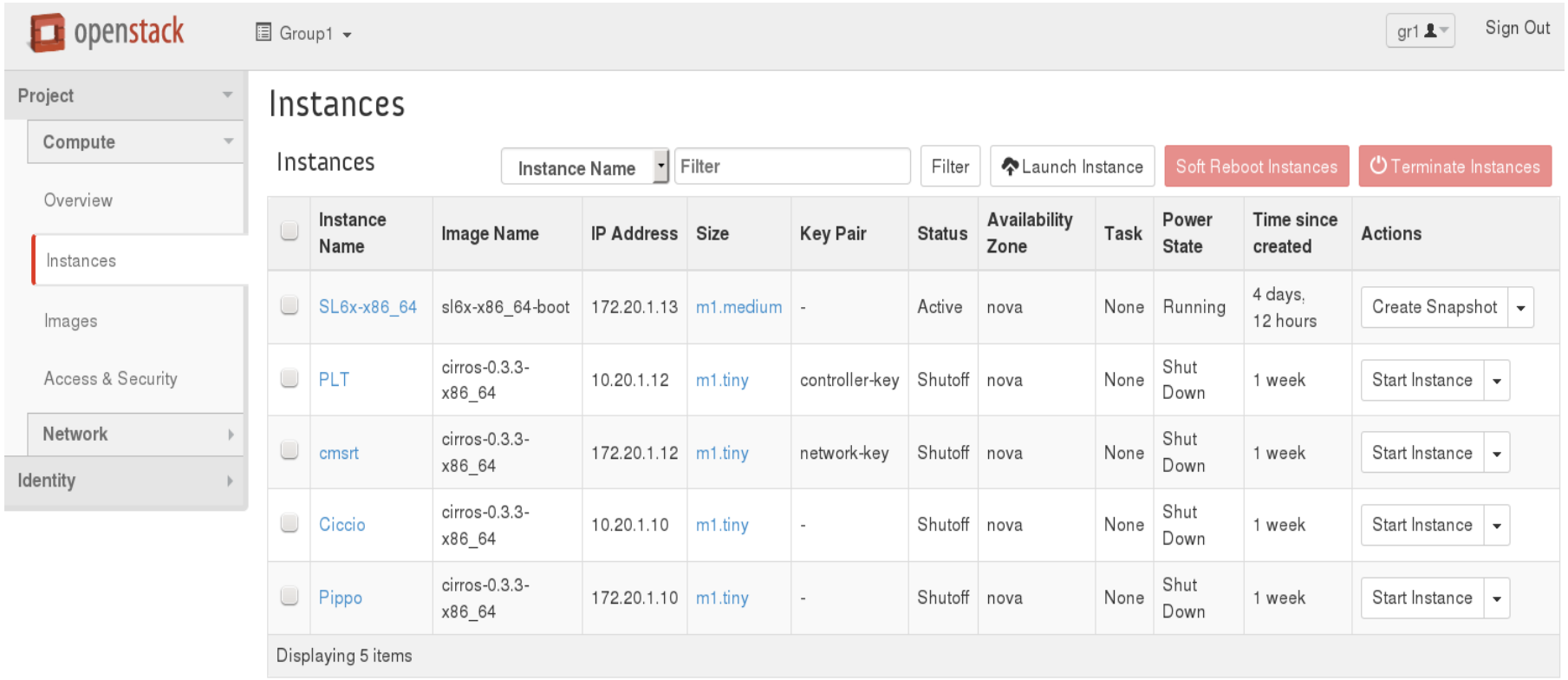
HORIZON (Dashboard)



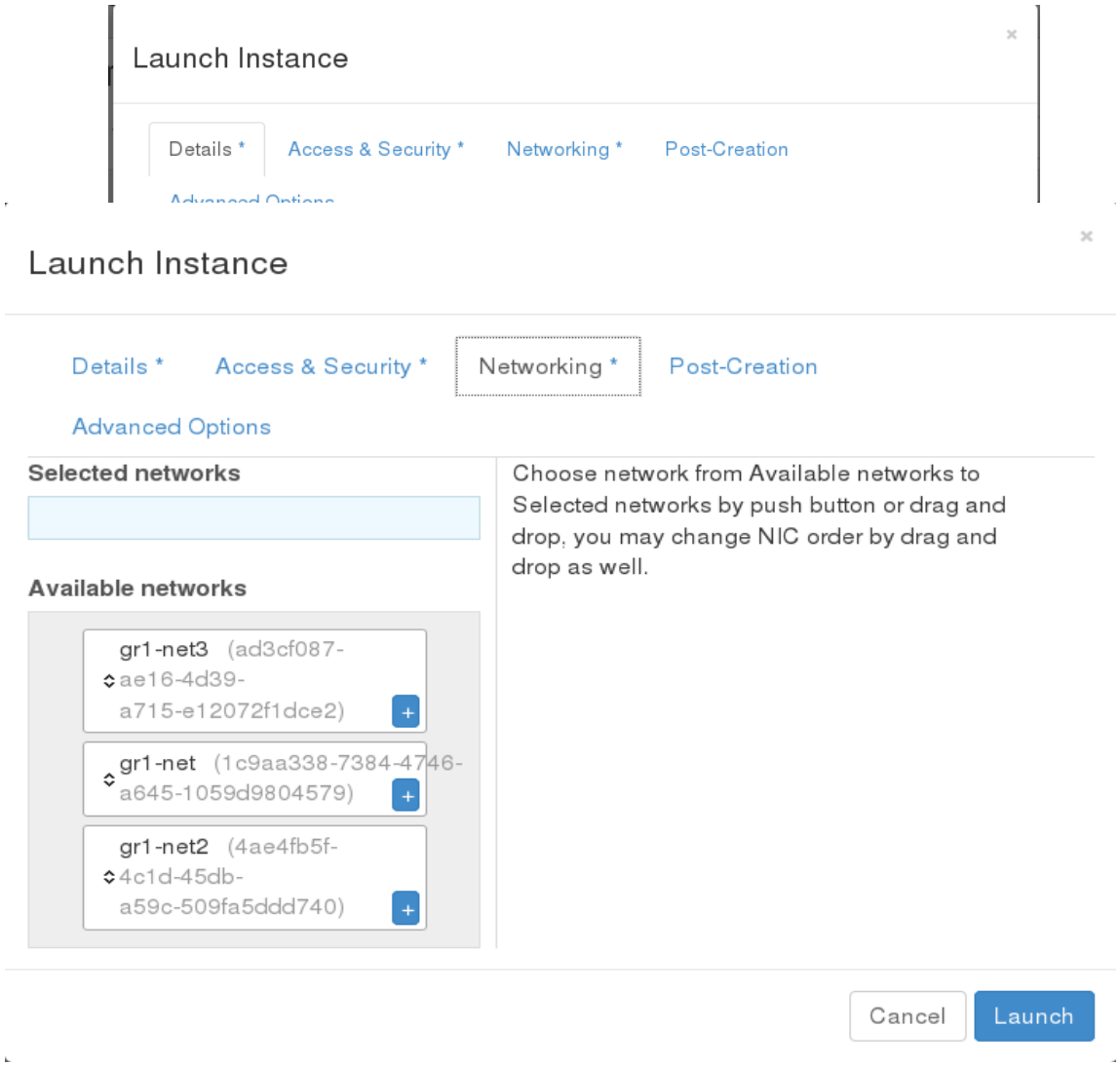
HORIZON (Dashboard)



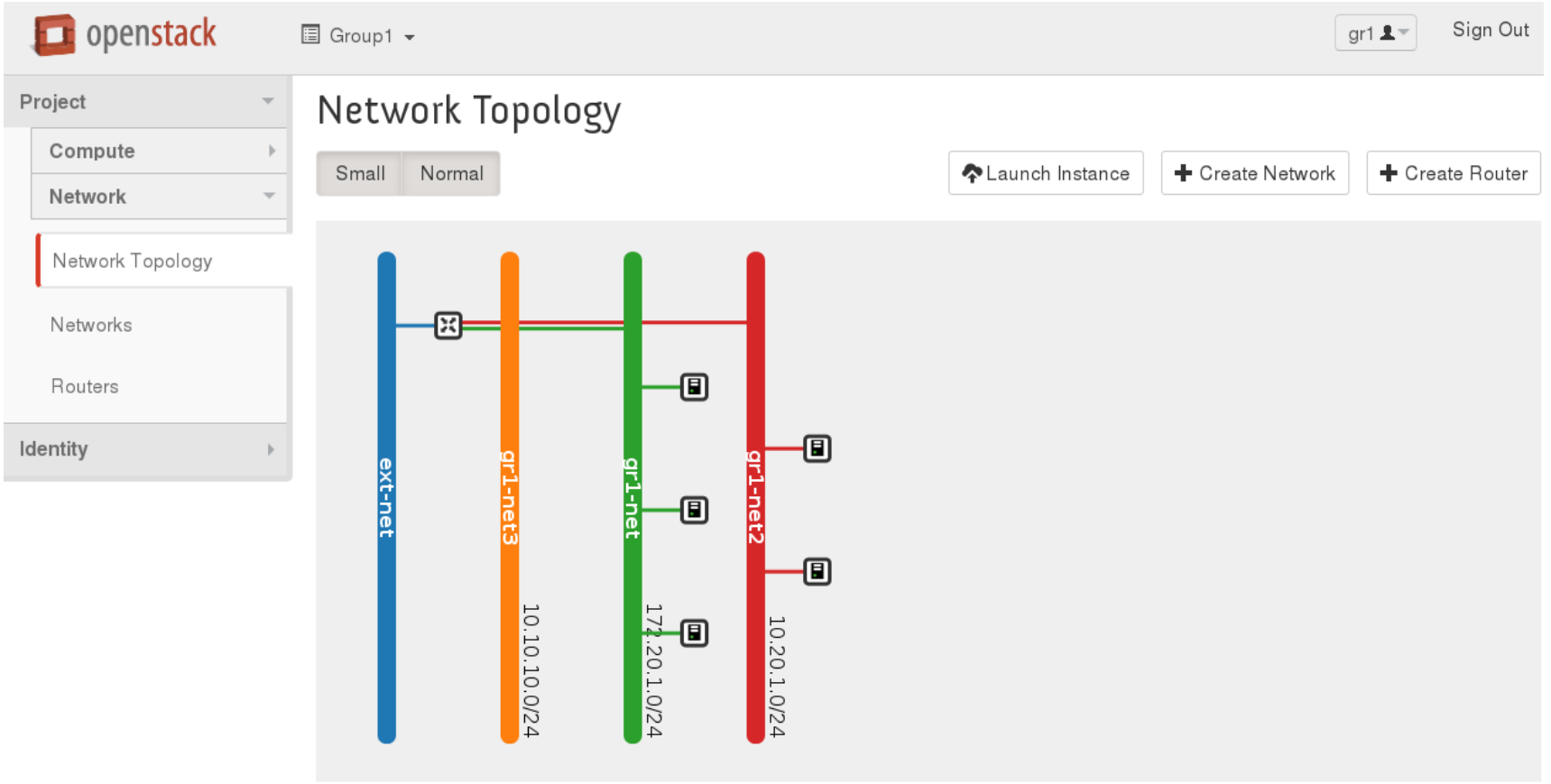
HORIZON (Dashboard)



HORIZON (Dashboard)



HORIZON (Dashboard)



Tipica installazione

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

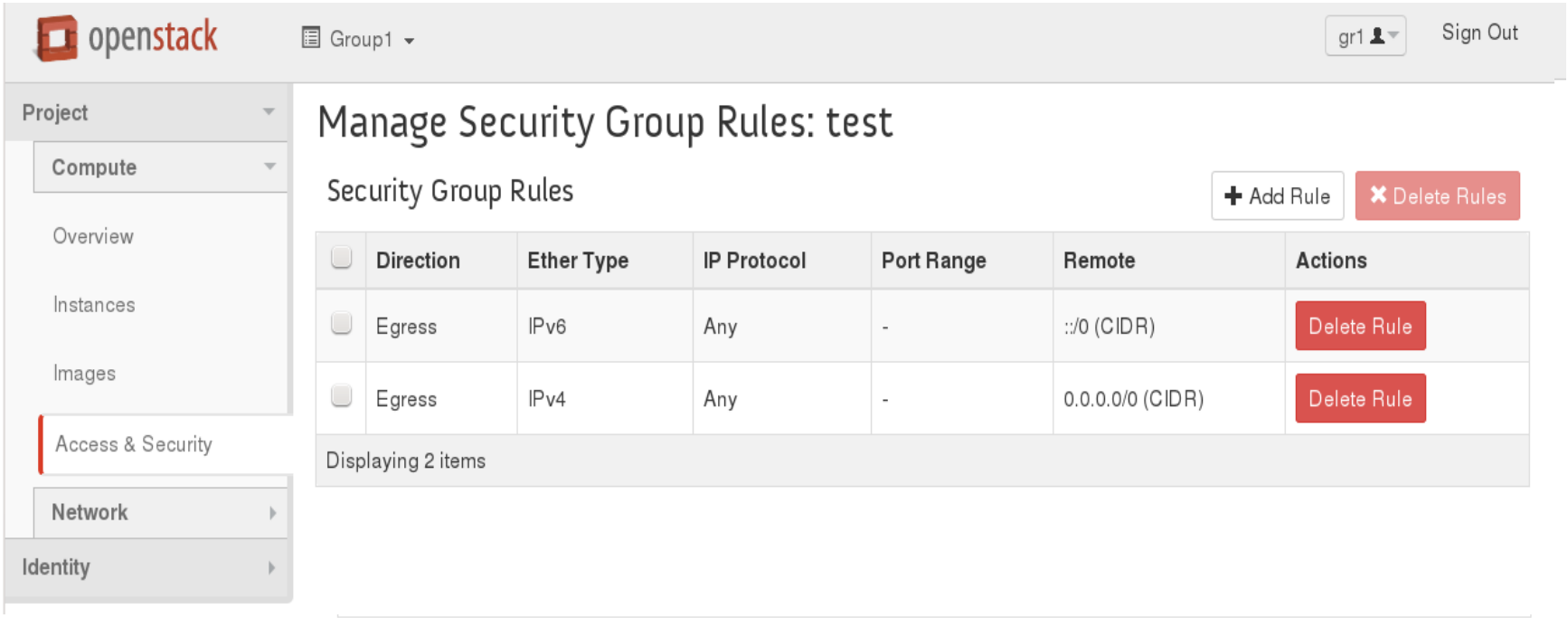
Istanze

- 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

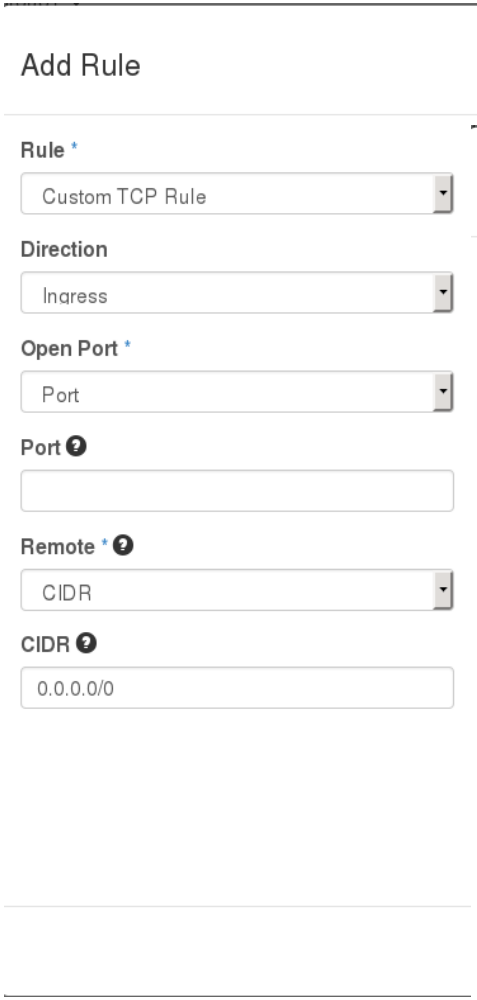
Security Groups

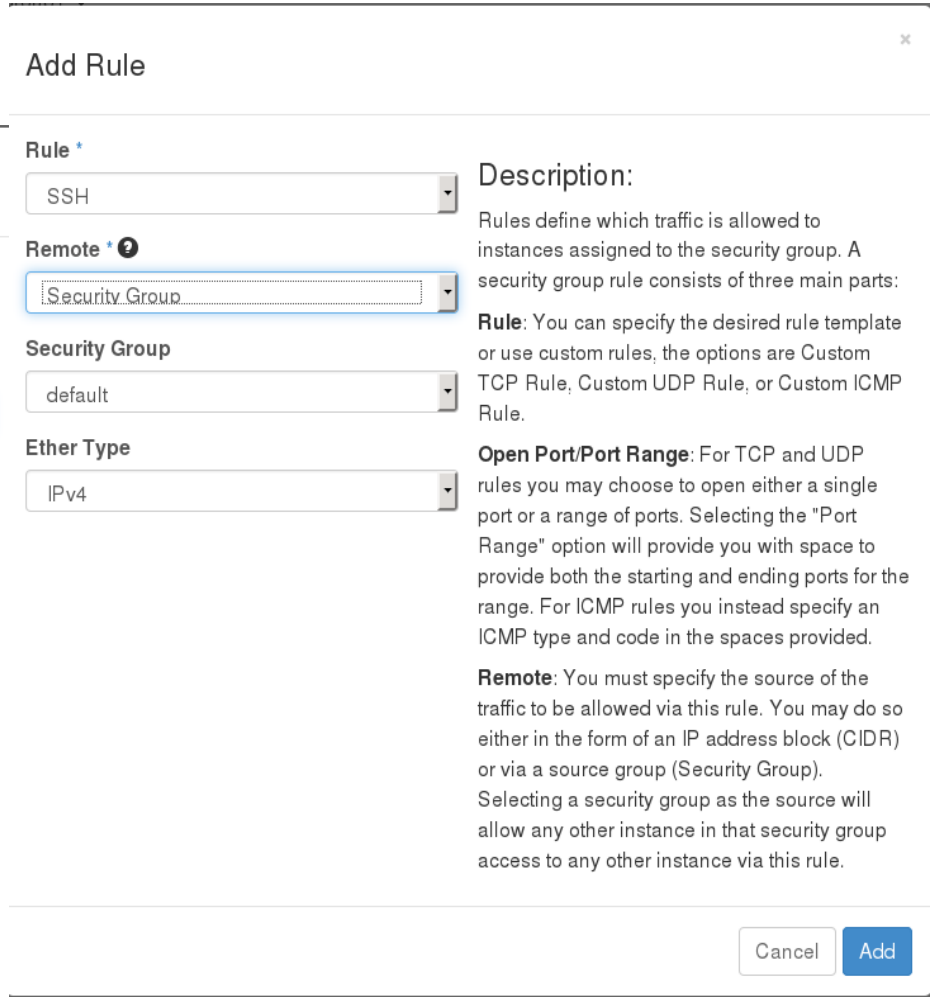
- sono un insieme di regole ip filter che vengono applicate sull’istanza in fase di creazione della stessa e possono essere modificate successivamente dall’utente amministratore del tenant
- se non viene specificato, di default viene applicato un security group che blocca il traffico
- i membri del tenant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

Security Groups

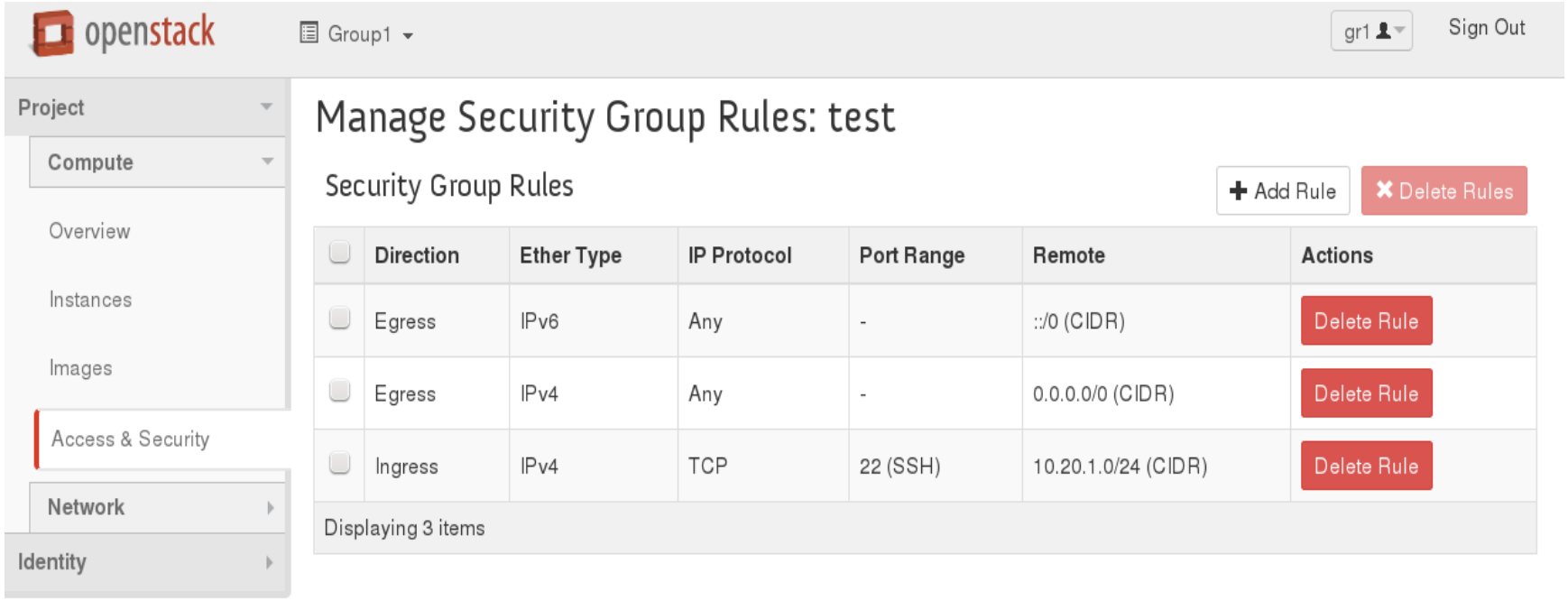


Security Groups

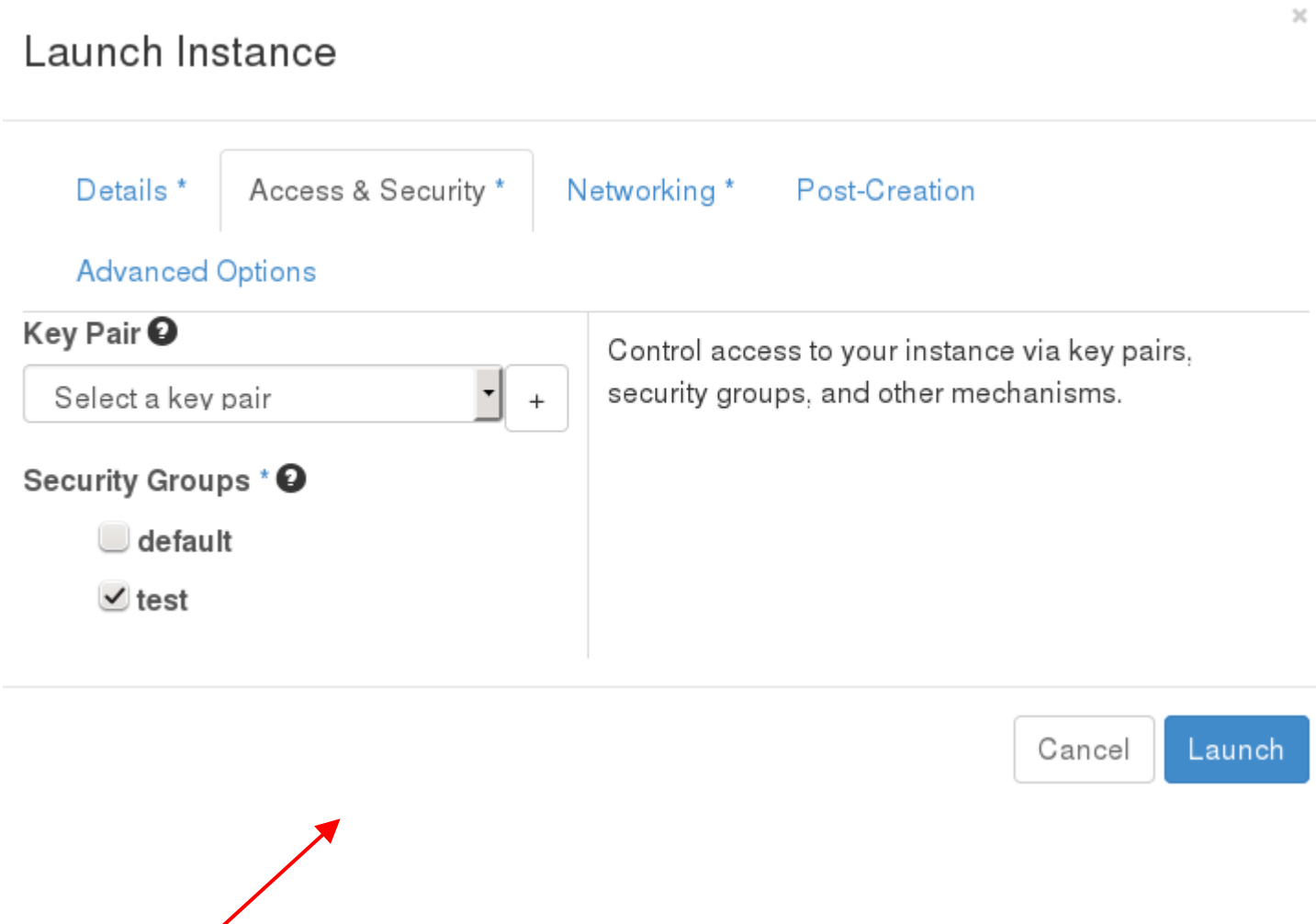




Security Groups



Security Groups



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/middleware-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