



Lietuvos mokslo ir studijų institucijų kompiuterių tinklas LITNET



Vilniaus universitetas

## **LITNET debesies infrastruktūrinio lygmens paslauga**

### **Paslaugos administravimo instrukcija**

Paslauga sukurta vykdant Europos socialinio fondo finansuojamą projektą „Mokslo ir studijų institucijoms LITNET teikiamų IT paslaugų plėtra“ Nr. 09.3.3-ESFA-V-711-01-0003



Kuriame  
Lietuvos ateitį

2014–2020 metų  
Europos Sąjungos  
fondų investicijų  
veiksmų programa

Vilnius

2019 m.

# Turinys

Įvadas.....	3
Pagrindiniai OpenStack komponentai: .....	4
Rekomendacijos dėl diegimo .....	5
MAAS diegimas.....	6
Juju diegimas.....	7
OpenStack diegimas.....	7
Nova Compute diegimas .....	7
Swift Storage .....	8
Neutron networking.....	8
Keystone.....	9
RabbitMQ.....	9
Nova Cloud Controller .....	9
OpenStack Dashboard .....	10
Glance .....	10
Ceph monitor.....	11
Cinder.....	11
NTP .....	11
Testavimai .....	12
Virtualios mašinos migravimas iš vienos DC OpenStack infrastruktūros į kitą DC OpenStack ..	13
Naudota literatūra .....	18

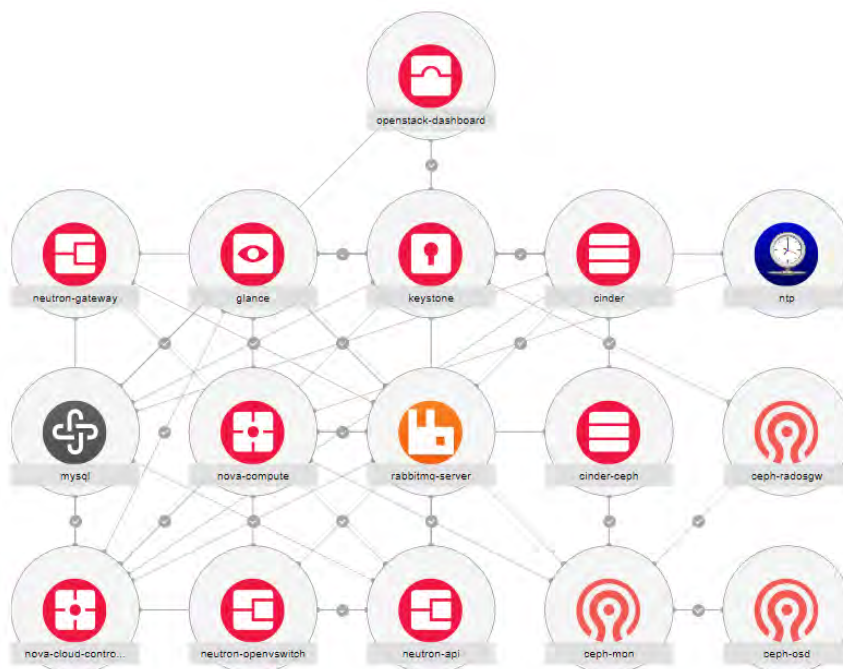
## **Įvadas**

Paslauga skirta mokslininkams ir tyrėjams, vykdančioms sudėtingus MTEP projektus, kuriuose reikalinga surinkti, saugoti bei apdoroti didelius duomenų kiekius ar valdyti dideles eksperimentines infrastruktūras. Paslauga taip pat gali būti aktuali ir MSI institucijų techninių centrų specialistams, kuriantiems naujas paslaugas mokslininkams ir tyrėjams.

Projekto veiklos įgyvendinimo metu trijuose LITNET duomenų centruose įdiegti OpenStack sprendimai. Įdiegtas OpenStack sprendimas leidžia eksporto/importo būdu migruoti virtualias mašinas iš vieno duomenų centro OpenStack hipervizoriaus į kito duomenų centro OpenStack hipervizorių. Tai sudaro galimybes dalintis LITNET duomenų centrų resursais, optimaliai išnaudojant turimus LITNET OpenStack išteklius ir teikti geriausiai atitinkančias naudotojo poreikius paslaugas.

LITNET debesies infrastruktūrinio lygmens paslaugą sudaro skaičiavimo, tinklo ir saugyklų resursai, kuriuos galima panaudoti kuriant sudėtingas mokslinio eksperimento valdymo ir didelio duomenų kiekio apdorojimo sistemas, debesijos paslaugų sistemas bei kitas informacines sistemas su galimybe jas valdyti per standartizuotą OpenStack API. Paslauga taip pat leidžia naudotis objektinėmis saugyklomis.

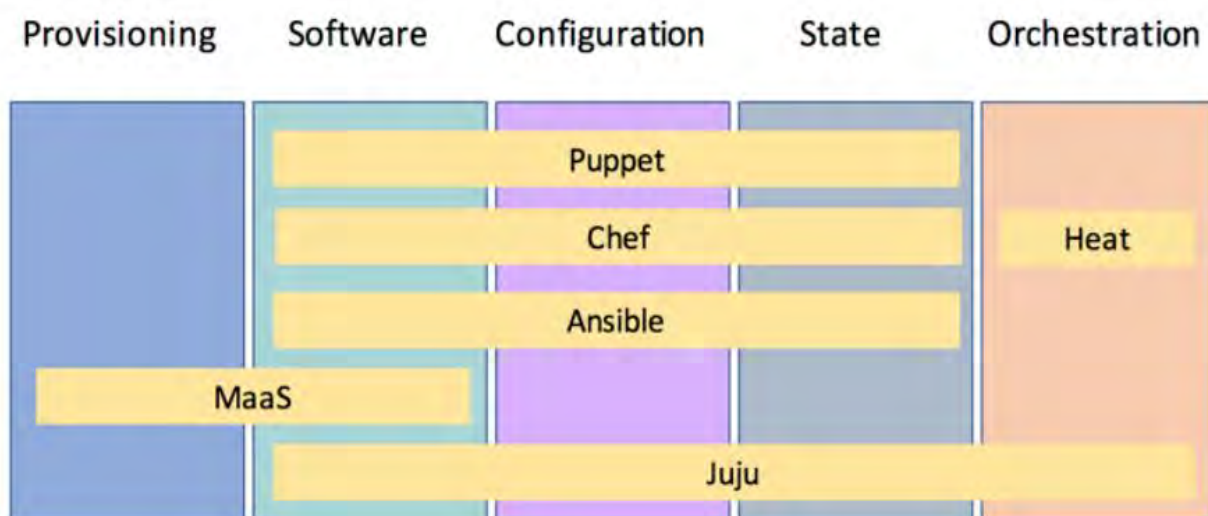
## Pagrindiniai OpenStack komponentai:



- Nova – pagrindinis OpenStack komponentas, kuris naudojamas virtualių mašinų diegimui ir valdymui, kartais gali atlikti skaičiavimo užduotis.
- Swift – komponentas atsakingas už duomenų saugojimą. Palengvina diegėjui užtikrinti saugomų duomenų saugoma, perimdama visą darbą sau.
- Cinder – swift galėjo būti failinė, blokinė saugykla, tai cinder gali būti tik blokinė saugojimo saugykla.
- Neutron – komponentas atsakingas už tinklo valdymą. Užtikrina, kad kiekvienas komponentas galės komunikuoti tinkle.
- Horizon – grafinė valdymo sąsaja, leidžia administratoriui valdyti virtualias mašinas, bei pasižiūrėti kas vyksta su visu Cloud.
- Keystone – atsakingas už identifikaciją OpenStack. Tai yra pagrindinis komponentas, kuri centralizuotai gali susieti naudotojų paskyras su teikiamomis Cloud paslaugomis.
- Glance – teikia virtualių mašinų atvaizdų (angl. Image) paslaugą. Leidžia atvaizdus naudoti naujų virtualių mašinų kūrimui.
- Ceilometer – tai yra telemetrijos komponentas, kuris atsakingas už paslaugų naudojimo skaičiavimus. Šis komponentas apskaičiuoja, kiek laiko, kuris naudotojas arba projektas naudojo resursus.
- Heat – orchestravimo komponentas, leidžiantis administratoriaus lengviau valdyti OpenStack, iš anksto pasiruošti konfigūracijos failus.

OpenStack sudarytas iš įvairių komponentų, sudiegti viską atskirai gali būti sudėtinga, todėl geriausia pasinaudoti automatizacijos įrankiais, kurie leidžia lengviau atlikti diegimus, kai prieš diegimą yra paruošiama visa konfigūracija. Automatizacijai rekomenduojame naudoti du įrankius: MAAS ir Juju. Įdiegiant fizinį serverį rekomenduojame naudoti MAAS įrankį, kuris leidžia valdyti fizinius serverius, paruošti juos Cloud diegimui. MAAS sugeba valdyti didelį kiekį fizinių serverių, iš kurių sukuria vieną didelį resursą. Mašinos gali būti automatiškai paaimamos ir naudojamos kaip įprasta virtuali mašina.

MAAS schema:



Juju yra atviro kodo programų modeliavimo įrankis, leidžiantis greitai, efektyviai diegti, konfigūruoti, valdyti debesų infrastruktūrą viešuose ir privačiuose debesyse. Leidžia pasiekti daugybę geriausios praktikos sprendimų, kuriuos galite įdiegti viena komanda. Juju naudoja įrankį Charms. Charms – yra diegimo instrukcijų rinkiniai, kuriuose yra aprašytos visos operacijos, reikalingos įdiegti, konfigūruoti ir prižiūrėti Cloud. Charms yra viena programa, kuri apima visą kodą bei žinias, kurių reikia norint valdyti Cloud.

## Rekomendacijos dėl diegimo

Minimalūs reikalavimai įrangai:

- 1 x MAAS Rack with Region controller: 8GB RAM, 2 CPUs, 1 NIC, 40GB storage
- 1 x Juju node: 4GB RAM, 2 CPUs, 1 NIC, 40GB storage
- 4 x OpenStack cloud nodes: 8GB RAM, 2 CPUs, 2 NICs, 80GB storage

## MAAS diegimas

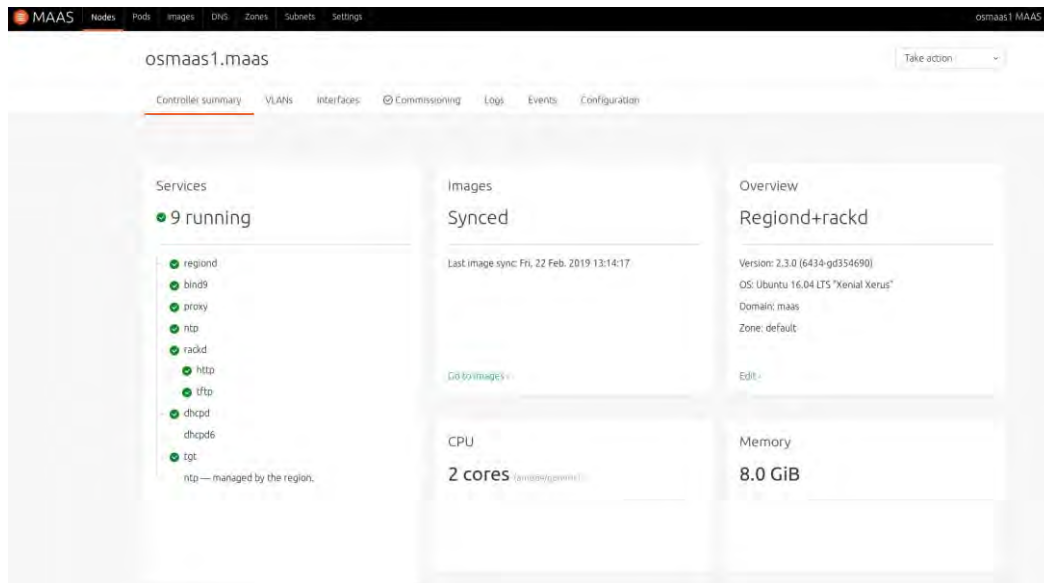
Įsidiegiame Ubuntu Server OS, įsidiegiame MAAS, po diegimo susikuriame admin naudotoją:

```
sudo apt install maas
```

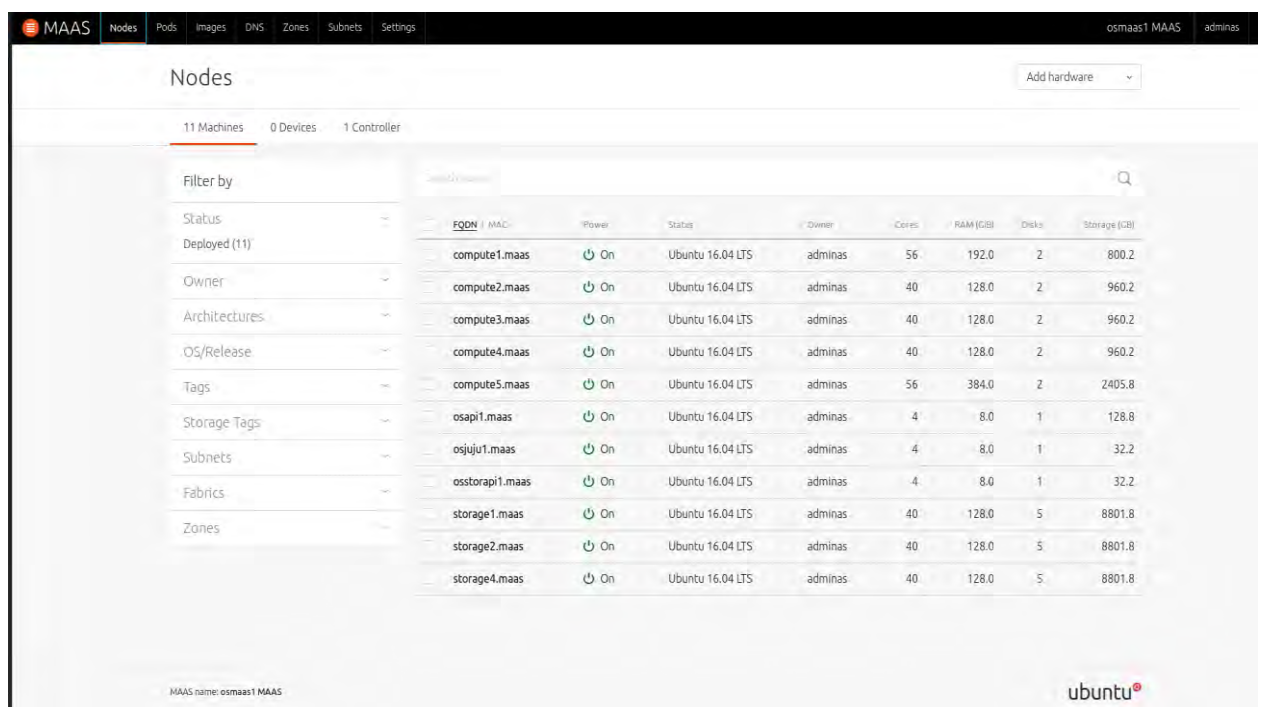
```
sudo maas createadmin
```

Sėkmingai įdiegę MAAS, jį galime pasiekti adresu <http://<your.maas.ip>:5240/MAAS/>.

Po diegimo reikės sukonfigūruoti tinklus, įkelti OS atvaizdus bei atlikti kitus veiksmus.



MAAS fizinių serverių sąrašas:



## **Juju diegimas**

```
sudo add-apt-repository -u ppa:juju/stable
```

```
sudo apt install juju
```

Sukuriame savo Cloud

```
juju add-cloud debesis
```

Po to vedame MAAS ir nurodome jo adresą <http://<your.maas.ip>:5240/MAAS/>

```
juju add-credential debesis
```

Sukuriame naudotoją, jeigu reikalinga, papildomai nurodome MAAS Keys iš MAAS puslapio.

```
juju bootstrap --constraints tags=juju debesis maas-controller
```

Jeigu viskas sėkmingai ekrane turi parašyti, kažką panašaus

```
Creating Juju controller "maas-controller" on debesis
```

```
Looking for packaged Juju agent version 2.2-alpha1 for amd64
```

```
Launching controller instance(s) on debesis...
```

```
- 7cm8tm (arch=amd64 mem=2G cores=2)
```

```
Fetching Juju GUI 2.4.4
```

```
Waiting for address
```

```
Attempting to connect to 192.168.100.106:22
```

```
Bootstrap agent now started
```

```
Contacting Juju controller at 192.168.100.106 to verify accessibility...
```

```
Bootstrap complete, "maas-controller" controller now available.
```

```
Controller machines are in the "controller" model.
```

```
Initial model "default" added.
```

## **OpenStack diegimas**

### **Nova Compute diegimas**

Paruošiamė failą compute.yaml

```
nova-compute:
```

```
enable-live-migration: True
```

```
enable-resize: True
```

```
migration-auth-type: ssh
```

```
virt-type: qemu
```

Pirmos mašinos diegimas

```
juju deploy --to 1 --config compute.yaml nova-compute
```

Jeigu reikia papildomai kitų mašinų, naudojame kitą komandą

```
juju add-unit --to 2 nova-compute
```

```
juju add-unit --to 3 nova-compute
```

Tikriname, ar viskas gerai įdiegta

```
juju status nova-compute
```

### **Swift Storage**

Paruošiamė failą swift-storage.yaml

```
swift-storage:
```

```
block-device: sdc
```

```
overwrite: "true"
```

Diegiamė

```
juju deploy --to 0 --config swift-storage.yaml swift-storage
```

```
juju add-unit --to 1 swift-storage
```

```
juju add-unit --to 2 swift-storage
```

```
juju add-unit --to 3 swift-storage
```

### **Neutron networking**

Paruošiamė failą neutron.yaml

```
neutron-gateway:
```

```
ext-port: 'eth1'
```

```
neutron-api:
```

```
neutron-security-groups: True
```

Diegiamė Gateway

```
juju deploy --to 0 --config neutron.yaml neutron-gateway
```

Vėliau



```
juju deploy --to lxd:1 --config neutron.yaml neutron-api
```

```
juju deploy neutron-openvswitch
```

Sukuriame ryšius

```
juju add-relation neutron-api neutron-gateway
```

```
juju add-relation neutron-api neutron-openvswitch
```

```
juju add-relation neutron-openvswitch nova-compute
```

## **Keystone**

Paruošiamo failą keystone.yaml

```
keystone:
```

```
admin-password: openstack
```

Diegame

```
juju deploy --to lxd:3 --config keystone.yaml keystone
```

Pridedame

```
juju add-relation keystone mysql
```

```
juju add-relation neutron-api keystone
```

## **RabbitMQ**

Šis servisas naudojamas pranešimų siuntimui.

```
juju deploy --to lxd:0 rabbitmq-server
```

Sukuriame sujungimus

```
juju add-relation neutron-api rabbitmq-server
```

```
juju add-relation neutron-openvswitch rabbitmq-server
```

```
juju add-relation nova-compute:amqp rabbitmq-server
```

```
juju add-relation neutron-gateway:amqp rabbitmq-server:amqp
```

## **Nova Cloud Controller**

Paruošiamo failą controller.yaml

```
nova-cloud-controller:
```

*network-manager: "Neutron"*

Diegame

*juju deploy --to lxd:2 --config controller.yaml nova-cloud-controller*

Sukuriame prisijungimus

*juju add-relation nova-cloud-controller mysql*

*juju add-relation nova-cloud-controller keystone*

*juju add-relation nova-cloud-controller rabbitmq-server*

*juju add-relation nova-cloud-controller neutron-gateway*

*juju add-relation neutron-api nova-cloud-controller*

*juju add-relation nova-compute nova-cloud-controller*

## **OpenStack Dashboard**

Diegame konsolę

*juju deploy --to lxd:3 openstack-dashboard*

Sukuriame prisijungimą su Keystone

*juju add-relation openstack-dashboard keystone*

## **Glance**

Diegimas:

*juju deploy --to lxd:2 glance*

Sujungimai:

*juju add-relation nova-cloud-controller glance*

*juju add-relation nova-compute glance*

*juju add-relation glance mysql*

*juju add-relation glance keystone*

*juju add-relation glance rabbitmq-server*

## **Ceph monitor**

Diegimas:

```
juju deploy --to lxd:1 ceph-mon
```

```
juju add-unit --to lxd:2 ceph-mon
```

```
juju add-unit --to lxd:3 ceph-mon
```

Sujungimai:

```
juju add-relation ceph-osd ceph-mon
```

```
juju add-relation nova-compute ceph-mon
```

```
juju add-relation glance ceph-mon
```

## **Cinder**

Paruošiamo failą cinder.yaml.

```
cinder:
```

```
glance-api-version: 2
```

```
block-device: None
```

Diegimas:

```
juju deploy --to lxd:1 --config cinder.yaml cinder
```

Sujungimai:

```
juju add-relation nova-cloud-controller cinder
```

```
juju add-relation cinder mysql
```

```
juju add-relation cinder keystone
```

```
juju add-relation cinder rabbitmq-server
```

```
juju add-relation cinder:image-service glance:image-service
```

```
juju add-relation cinder ceph-mon
```

## **NTP**

Diegimas:

```
juju deploy ntp
```

Sujungimai:

```
juju add-relation neutron-gateway ntp
```

```
juju add-relation ceph-osd ntp
```

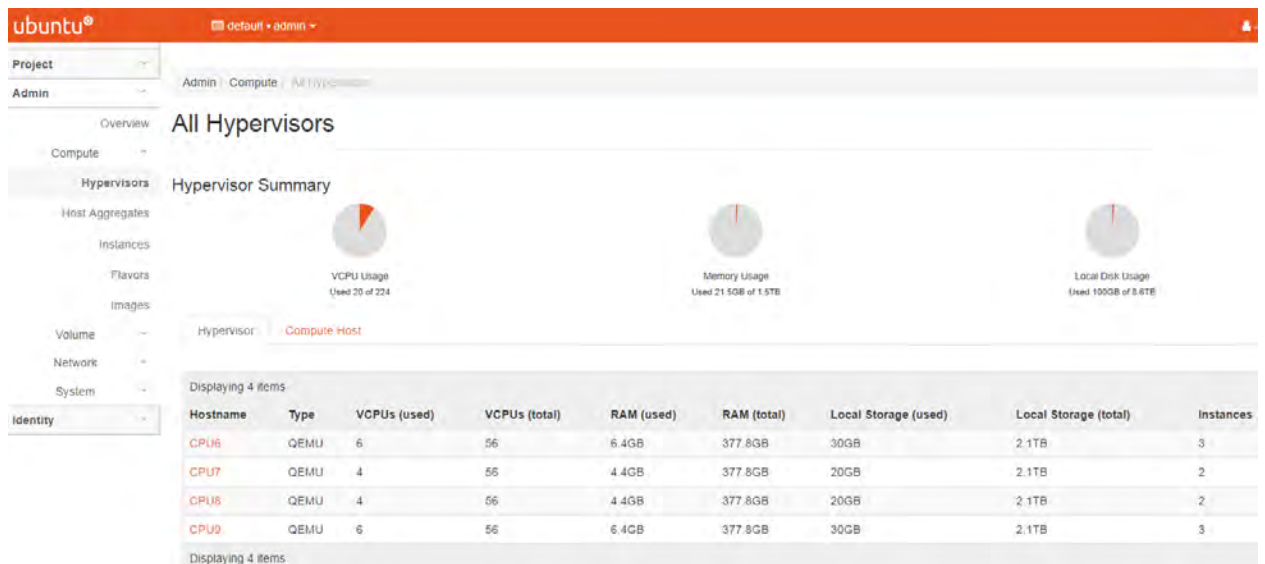
## Testavimai

```
juju status --format=yaml openstack-dashboard | grep public-address | awk '{print $2}'
```

Bandome atidaryti puslapį <http://<IP ADDRESS>/horizon>



Prisijungę turime pamatyti tokį puslapį.



Hostname	Type	VCPUs (used)	VCPUs (total)	RAM (used)	RAM (total)	Local Storage (used)	Local Storage (total)	Instances
CPU6	QEMU	6	56	6.4GB	377.8GB	30GB	2.1TB	3
CPU7	QEMU	4	56	4.4GB	377.8GB	20GB	2.1TB	2
CPU8	QEMU	4	56	4.4GB	377.8GB	20GB	2.1TB	2
CPU9	QEMU	6	56	6.4GB	377.8GB	30GB	2.1TB	3

Dar galime komandinės eilutės pagalba peržiūrėti būseną

```

nova-cloud-controller 14.0.10 active 1 nova-cloud-controller jujucharms 301 ubuntu
nova-compute 14.0.10 active 5 nova-compute jujucharms 274 ubuntu
ntp 4.2.8p4+dfsg active 6 ntp jujucharms 24 ubuntu
openstack-dashboard 10.0.5 active 1 openstack-dashboard jujucharms 251 ubuntu
rabbitmq-server 3.5.7 active 1 rabbitmq-server jujucharms 68 ubuntu

Unit Workload Agent Machine Public address Ports Message
ceph-mon/0 active idle 6/lxd/0 10.100.254.245 Unit is ready and clustered
ceph-mon/2 active idle 8/lxd/0 10.100.254.247 Unit is ready and clustered
ceph-mon/5* active idle 17/lxd/0 10.100.255.7 Unit is ready and clustered
ceph-osd/0 active idle 6 158.129.213.10 Unit is ready (4 OSD)
ceph-osd/2* active idle 8 158.129.213.11 Unit is ready (4 OSD)
ceph-osd/7 active idle 17 158.129.213.13 Unit is ready (4 OSD)
ceph-radosgw/0* active idle 1/lxd/0 10.100.254.235 80/tcp Unit is ready
cinder/0* active idle 1/lxd/1 10.100.254.239 8776/tcp Unit is ready
cinder-ceph/0* active idle 10.100.254.239 Unit is ready
glance/0* active idle 0/lxd/0 10.100.254.226 9292/tcp Unit is ready
keystone/0* active idle 0/lxd/1 10.100.254.229 5000/tcp Unit is ready
mysql/0* active idle 0/lxd/2 10.100.254.231 3306/tcp Unit is ready
neutron-api/0* active idle 0/lxd/3 10.100.254.230 9696/tcp Unit is ready
neutron-gateway/0* active idle 0 158.129.213.3 Unit is ready
ntp/0* active idle 158.129.213.3 123/udp Ready
nova-cloud-controller/0* active idle 1/lxd/2 10.100.254.237 8774/tcp Unit is ready
nova-compute/9* active idle 19 158.129.213.8 Unit is ready
neutron-openvswitch/7 active idle 158.129.213.8 Unit is ready
ntp/8 active idle 158.129.213.8 123/udp Ready
nova-compute/10 active idle 20 158.129.213.7 Unit is ready
neutron-openvswitch/9 active idle 158.129.213.7 Unit is ready
ntp/10 active idle 158.129.213.7 123/udp Ready
nova-compute/11 active idle 21 158.129.213.12 Unit is ready
neutron-openvswitch/8 active idle 158.129.213.12 Unit is ready
ntp/9 active idle 158.129.213.12 123/udp Ready
nova-compute/12 active idle 22 158.129.213.6 Unit is ready
neutron-openvswitch/10* active idle 158.129.213.6 Unit is ready
ntp/11 active idle 158.129.213.6 123/udp Ready
nova-compute/14 active idle 24 158.129.213.14 Unit is ready
neutron-openvswitch/11 active idle 158.129.213.14 Unit is ready
ntp/12 active idle 158.129.213.14 123/udp Ready
openstack-dashboard/0* active idle 0/lxd/4 10.100.254.225 80/tcp,443/tcp Unit is ready
rabbitmq-server/0* active idle 0/lxd/5 10.100.254.234 5672/tcp Unit is ready

Machine State DNS Inst id Series AZ Message
0 started 158.129.213.3 0ggqns xenial default Deployed
0/lxd/0 started 10.100.254.226 juju-5e7e2f-0-lxd-0 xenial default Container started
0/lxd/1 started 10.100.254.229 juju-5e7e2f-0-lxd-1 xenial default Container started
0/lxd/2 started 10.100.254.231 juju-5e7e2f-0-lxd-2 xenial default Container started
0/lxd/3 started 10.100.254.230 juju-5e7e2f-0-lxd-3 xenial default Container started
0/lxd/4 started 10.100.254.225 juju-5e7e2f-0-lxd-4 xenial default Container started
0/lxd/5 started 10.100.254.234 juju-5e7e2f-0-lxd-5 xenial default Container started
1 started 158.129.213.5 m8hwwx xenial default Deployed
1/lxd/0 started 10.100.254.235 juju-5e7e2f-1-lxd-0 xenial default Container started
1/lxd/1 started 10.100.254.239 juju-5e7e2f-1-lxd-1 xenial default Container started
1/lxd/2 started 10.100.254.237 juju-5e7e2f-1-lxd-2 xenial default Container started

```

## Virtualios mašinos migravimas iš vienos DC OpenStack infrastruktūros į kitą DC OpenStack

Virtuali mašina išjungžiama. Informacija apie norimą eksportuoti virtualią mašiną perduodama OpenStack administratoriui. Šią informaciją galima rasti spragtelėjus ant virtualios mašinos (pavyzdžio atveju tai VM pavadinimu „Sistema migravimui“) OpenStack portale:

# Sistema migravimui

Overview	Log	Console	Action Log
<b>Name</b>	Sistema migravimui		
<b>ID</b>	441074ac-39b7-4798-9841-5218f152dfe5		
<b>Status</b>	Active		
<b>Locked</b>	False		
<b>Availability Zone</b>	cinderLVM		
<b>Created</b>	21 May 2019, 10:52 a.m.		
<b>Time Since Created</b>	17 hours, 47 minutes		
<b>Host</b>	ostack-ibm2		

## Specs

<b>Flavour Name</b>	Mini
<b>Flavour ID</b>	0cf02fe6-5415-4f8d-9fcc-d6db9988103d
<b>RAM</b>	512MB
<b>VCPUs</b>	1 VCPU
<b>Disk</b>	2GB

## IP Addresses

<b>Internal</b>	10.0.0.13
-----------------	-----------

## Security Groups

<b>default</b>	ALLOW IPv6 from default ALLOW IPv4 to 0.0.0.0/0 ALLOW IPv4 from default ALLOW IPv4 1-65535/tcp from 0.0.0.0/0 ALLOW IPv4 icmp to 0.0.0.0/0 ALLOW IPv6 to ::/0 ALLOW IPv4 1-65535/tcp to 0.0.0.0/0 ALLOW IPv4 1-65535/udp from 0.0.0.0/0 ALLOW IPv4 icmp from 0.0.0.0/0 ALLOW IPv4 1-65535/udp to 0.0.0.0/0
----------------	---

## Metadata

<b>Key Name</b>	None
<b>Image</b>	None

## Volumes Attached

<b>Attached To</b>	0342d86b-9388-4bab-bad3-7f1ed45a7908 on /dev/vda
--------------------	--

Openstack administratorius suranda pilną VM atvaizdo ID paleidimo komandą OpenStack hipervizoriuje:

```
root@ostack-ibm2:~# rbd -p cinder-volumes --id ostack ls | grep 0342d86b-9388-4bab-bad3-7f1ed45a7908
volume-0342d86b-9388-4bab-bad3-7f1ed45a7908
```

Eksportuojamas VM atvaizdas iš ceph į laikiną katalogą:

```
root@ostack-ibm2:~# rbd --id ostack export --pool=cinder-volumes volume-0342d86b-9388-4bab-bad3-7f1ed45a7908 /temp_folder/volume-0342d86b-9388-4bab-bad3-7f1ed45a7908.img
Exporting image: 100% complete...done.
```

Eksportuotas atvaizdas perduodamas naudotojui jam patogiu būdu (pvz.: filesender.vu.lt).

Naudotojas, prisijungęs prie reikiamo duomenų centro OpenStack portalo, užaina į Compute -> Images, spaudžia mygtuką „Create image“. Atsivėrusiame lange įvedamas atvaizdo pavadinimas,

kaip atvaizdo šaltinis įkeliamas anksčiau eksportuotas VM atvaizdas, pasirenkamas atvaizdo formatas „Raw“ ir spaudžiamas mygtukas „Create Image“

Create Image

**Image Details**

Specify an image to upload to the Image Service.

**Image Name\***  
volume-0342d86b-9388-4bab-bad3-7f1

**Image Description**

**Image Source**

**Source Type**  
File

**File\***  
Browse... volume-0342d86b-9388-4t

**Format\***  
Raw

**Image Requirements**

**Kernel**  
Choose an image

**Ramdisk**  
Choose an image

**Architecture**

**Minimum Disk (GB)**  
0

**Minimum RAM (MB)**  
0

**Image Sharing**

**Visibility**  
Public Private

**Protected**  
Yes No

Cancel < Back Next > Create Image

Gali tekti palaukti , kol atvaizdas bus importuotas.

Kai atvaizdas baigiamas importuoti ir tvarkyti, iš jo galima kurti naują VM. Compute->Instances, skiltyje spaudžiamas mygtukas „Launch Instance“, atsivėrusiame lange įvedamas VM pavadinimas:

Launch Instance

Please provide the initial hostname for the instance, the availability zone where it will be deployed, and the instance count. Increase the Count to create multiple instances with the same settings.

**Instance Name \*** Sistema migravimuij

**Availability Zone** nova

**Count \*** 1

Total Instances (15 Max)

27%

3 Current Usage  
1 Added  
11 Remaining

Cancel < Back Next > Launch Instance

Kaip atvaizdo šaltinis nurodomas neseniai importuotas atvaizdas

Launch Instance

Instance source is the template used to create an instance. You can use a snapshot of an existing instance, an image, or a volume (if enabled). You can also choose to use persistent storage by creating a new volume.

**Select Boot Source** Image

**Create New Volume** Yes No

**Volume Size (GB) \*** 11

**Delete Volume on Instance Delete** Yes No

Allocated

Name	Updated	Size	Type	Visibility
> volume-0342d86b-9388-4bab-bad 3-7f1ed45a7908	5/22/19 9:14 AM	10.00 GB	raw	Private

Pasirenkami kiti nustatymai pagal poreikį.

Galiausiai spaudžiama „Launch instance“. Po šio žingsnio sukuriamas ir paleidžiamas VM.



## 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"/>	Sistema migravimui	-	192.168.54.7	VSP1	-	Active	nova	None	Running	0 minutes	Create Snapshot

Pirmas VM paleidimas gali užtrukti keletą minučių.

## Naudota literatūra

- <http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf>
- <http://www.datacenterknowledge.com/archives/2015/01/30/survey-half-of-private-clouds-are-openstack-clouds/>
- <https://www.openstack.org/software/project-navigator/>
- <http://ceph.com/>