Creating new servers

Once you have an account in Cleura Cloud, you can create virtual machines — henceforth simply *servers* — using either the Cleura Cloud Management Panel or the OpenStack CLI. Let us demonstrate the creation of a new server, following both approaches.

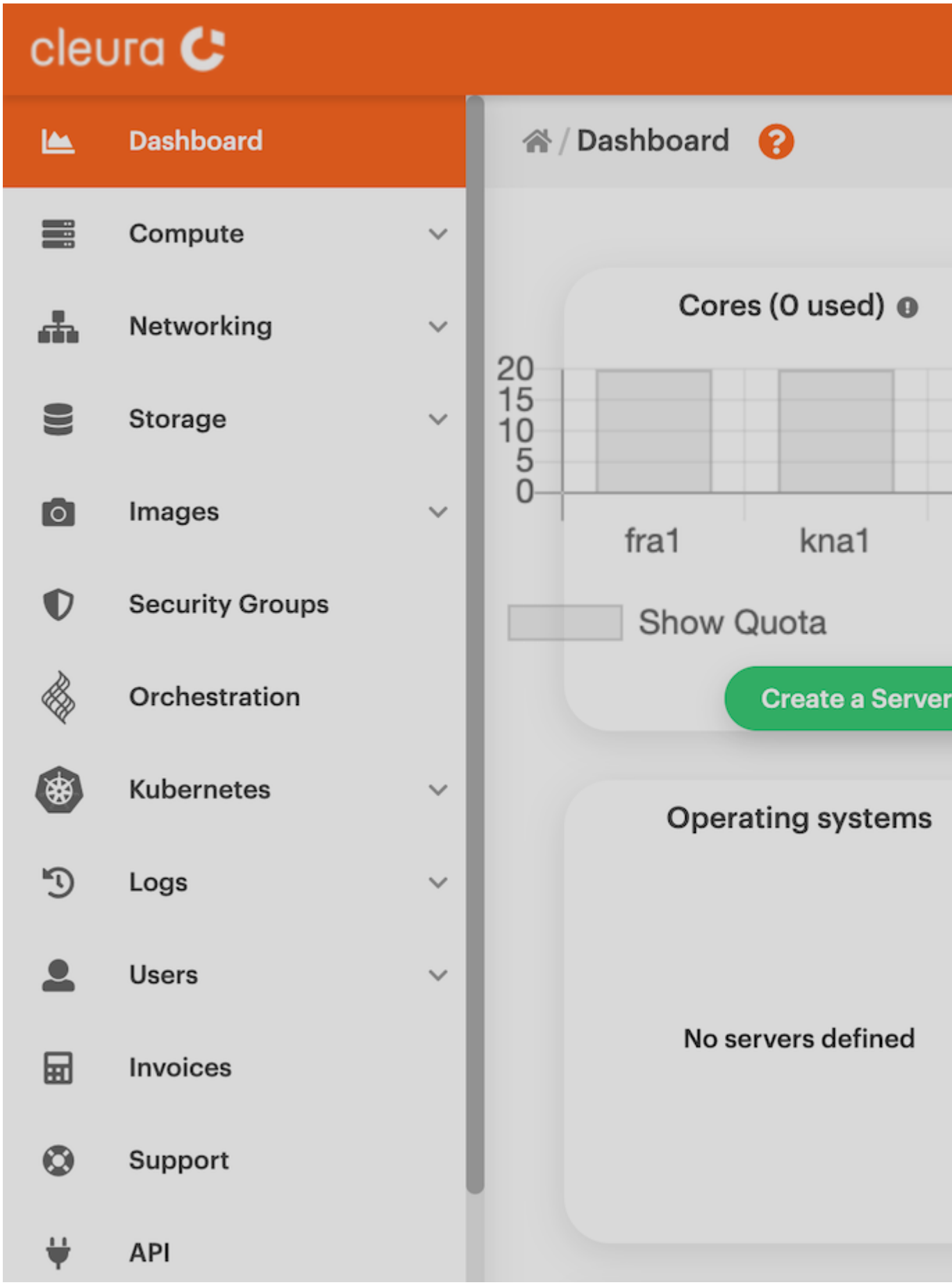
Prerequisites

You need to have at least one network in the region you are interested in. Additionally, if you prefer to work with the OpenStack CLI, then make sure to properly enable it first.

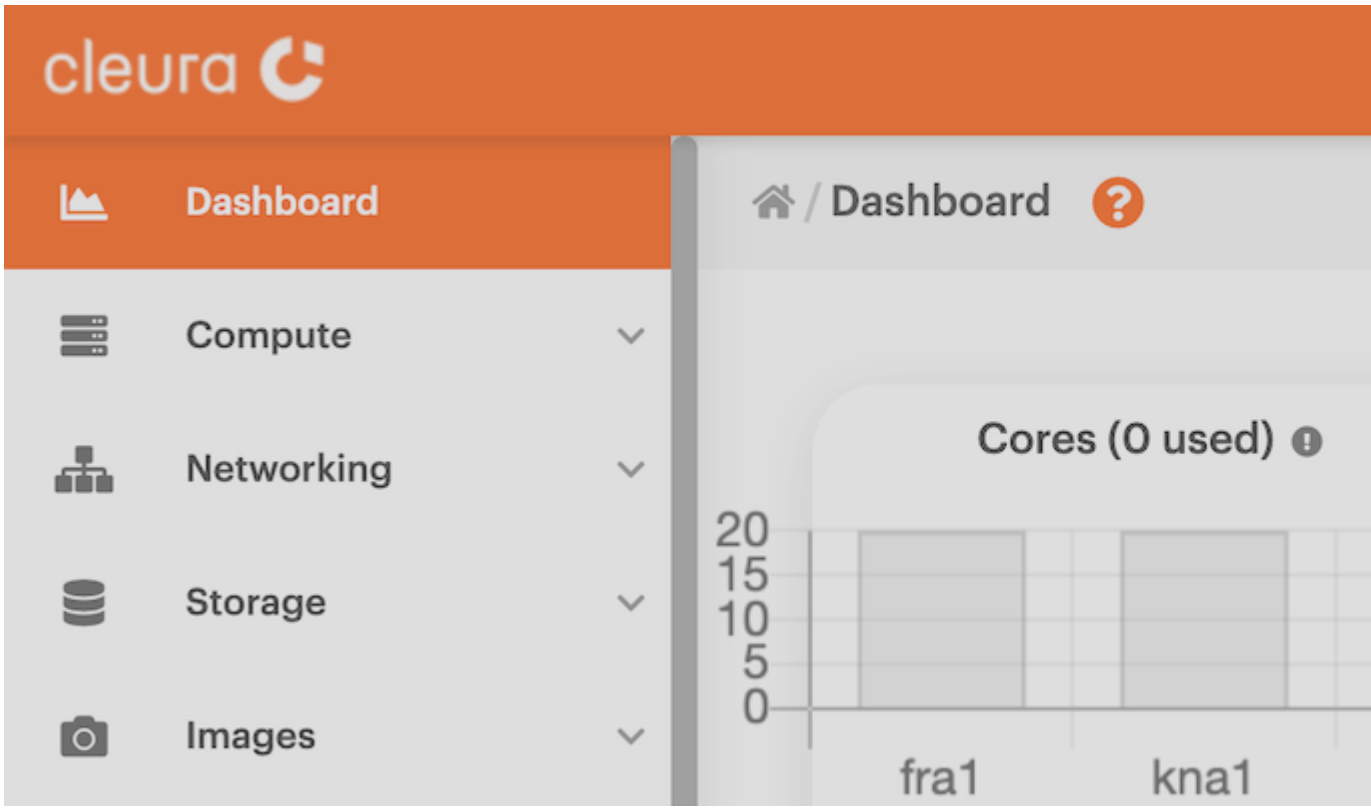
Creating a server

To create a server from the Cleura Cloud Management Panel, fire up your favorite web browser, navigate to the Cleura Cloud page, and log into your Cleura account. On the other hand, if you prefer to work with the OpenStack CLI, please do not forget to source the RC file first.

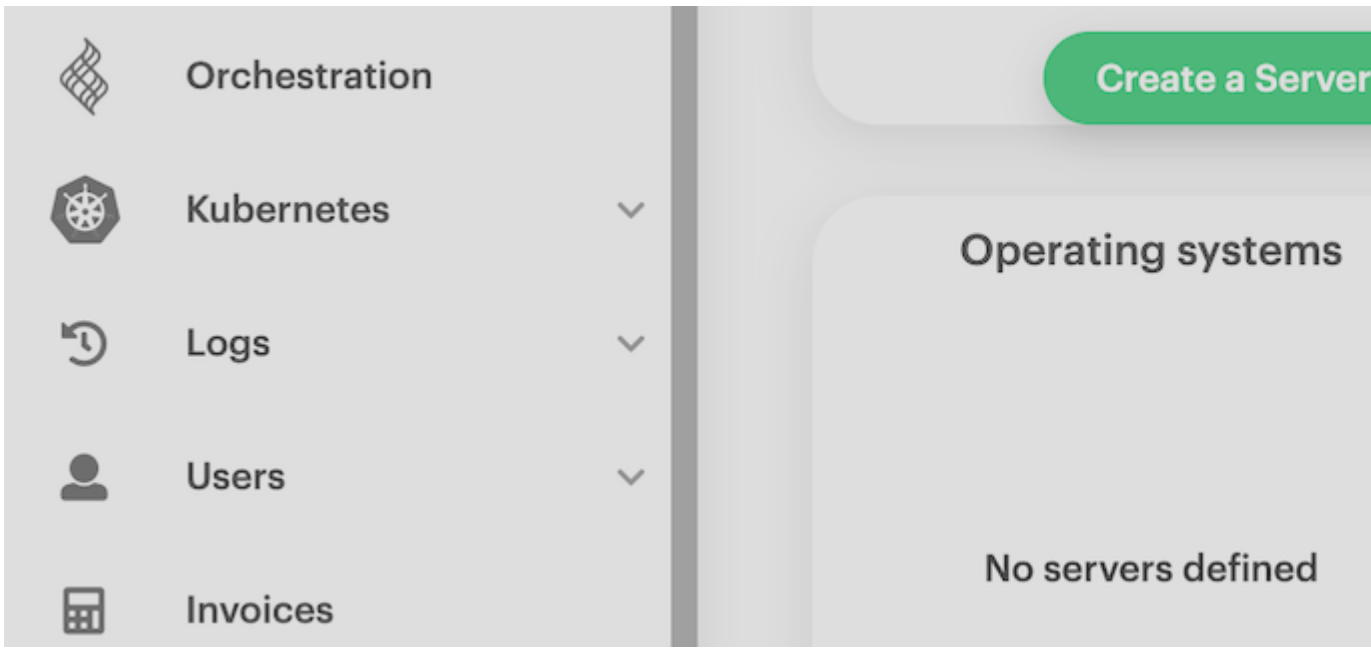
On the top right-hand side of the Cleura Cloud Management Panel, click the *Create* button. A new


cleura
Dashboard
Compute
Networking
Storage
Images
Security Groups
Orchestration
Kubernetes
Logs
Users
Invoices
Support
API
Dashboard
Cores (0 used)
20
15
10
5
0
fra1
kna1
Show Quota
Create a Server
Operating systems
No servers defined

You will notice several rounded boxes on that pane, each for defining, configuring, and instantiatin
one of the available regions.



In the *Boot source* section below, click the dropdown menu on the left and make sure you select *Bo*
this how-to guide, we have chosen *ubuntu* in general and *Ubuntu 22.04 Jammy Jellyfish 20220810*

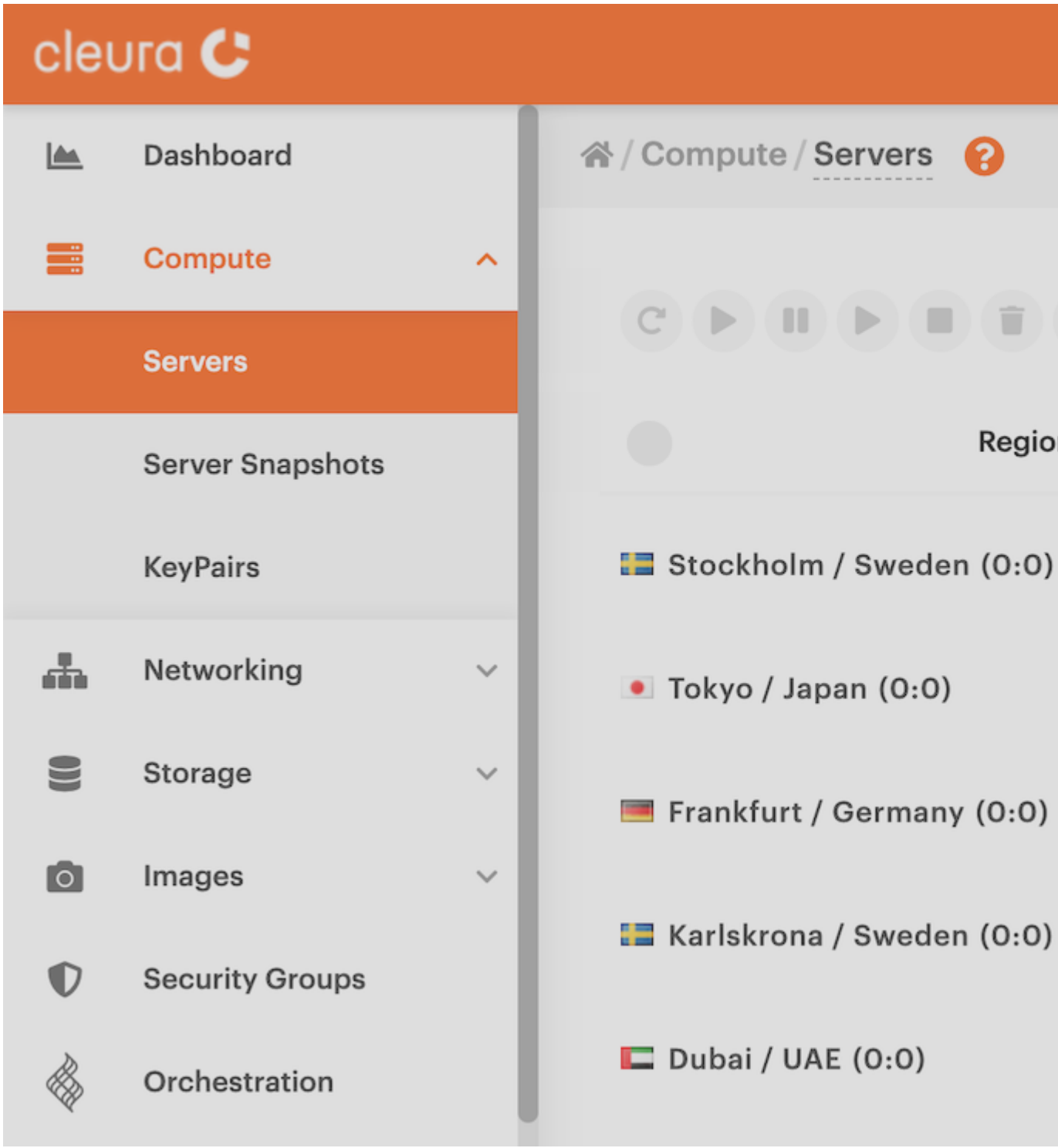


Next, make sure *Boot Target* is set to *Volume (Recommended)*. Regarding the server's CPU core co
configuration by clicking the dropdown menu at the right of *Flavor*. Please note that, depending on

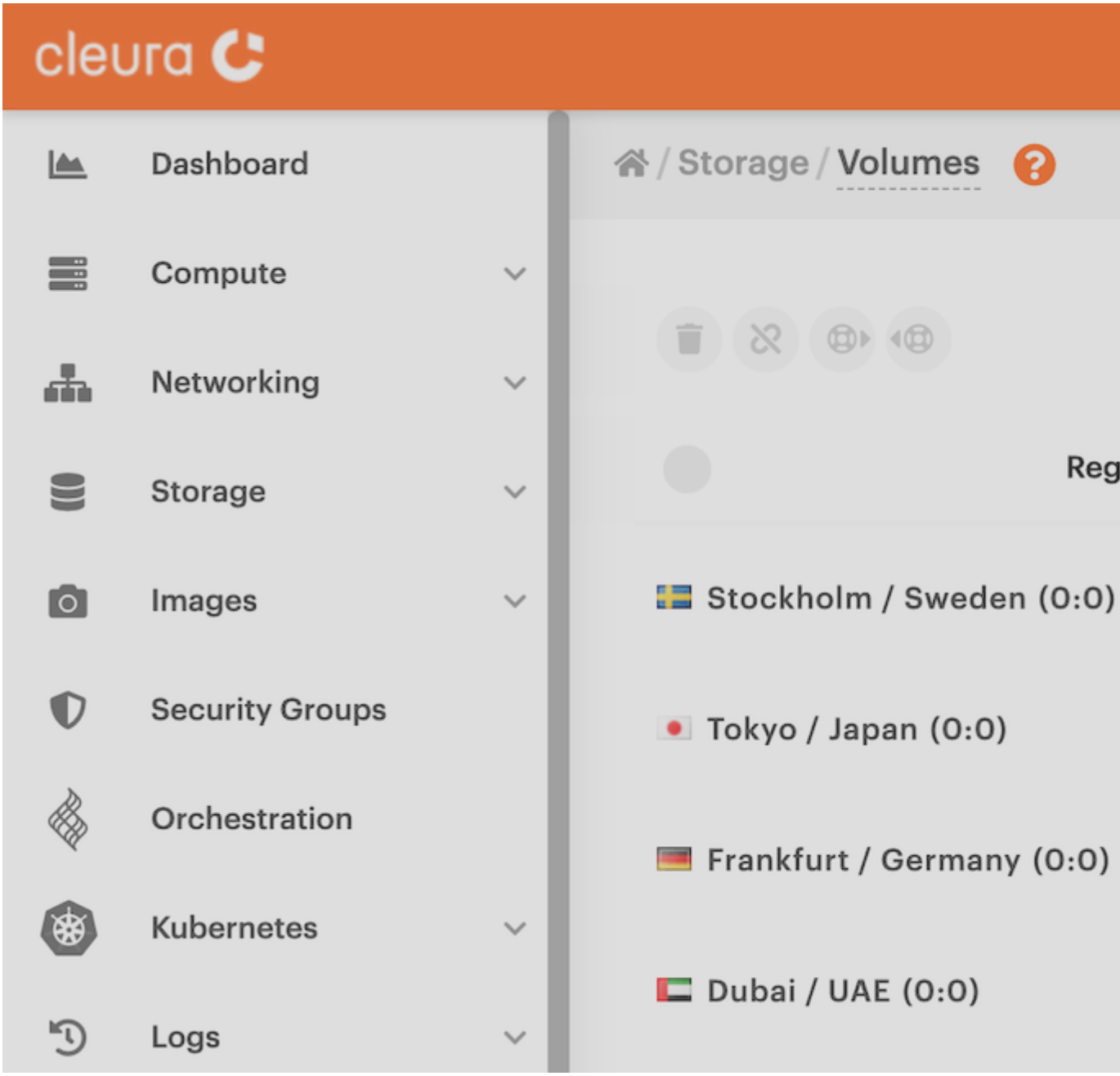
displayed in the green rectangular area at the top. Something else that affects the cost is the size o
than the default.

When, at a later time, you decide to delete the server, you can do so but **keep** its boot volume (you
hand, if you want your root volume to be automatically deleted when the server terminates, enable

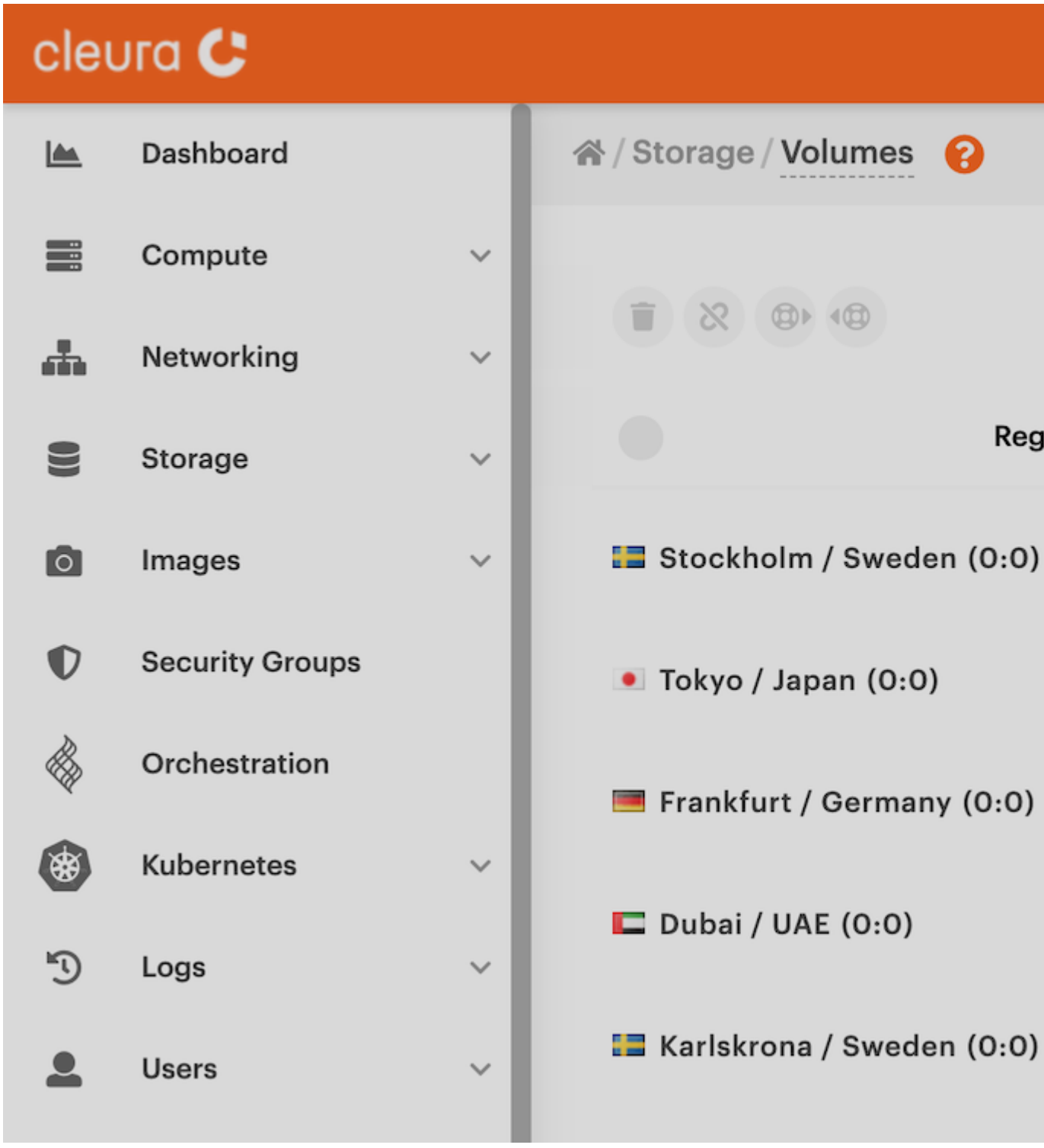
Finally, you may enable *Disaster recovery* for the server. If you do, then daily server snapshots will
(again, it is displayed in the green rectangular area at the top).



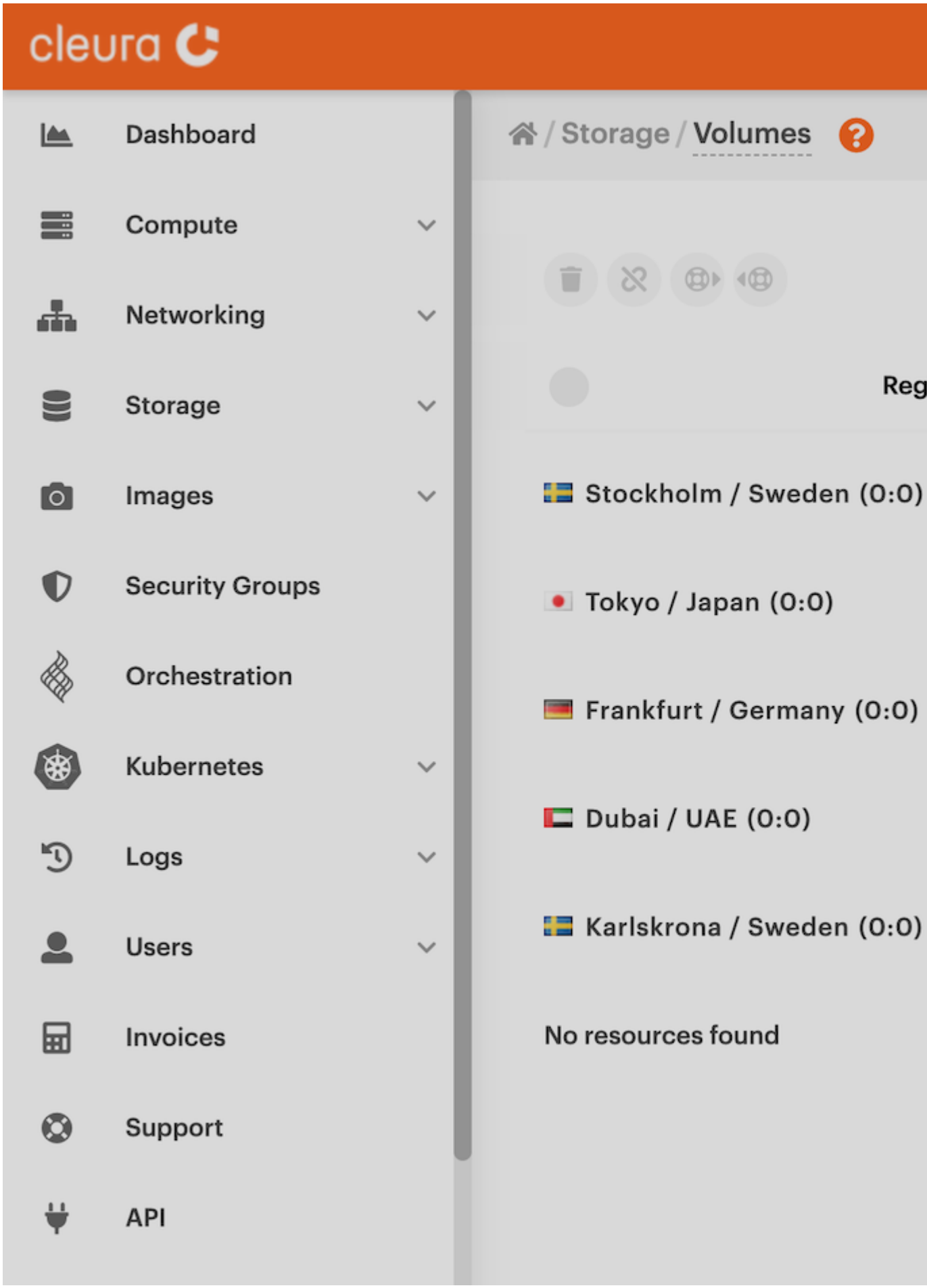
Regarding networking, select one of the available networks to attach the new server to. If you wan



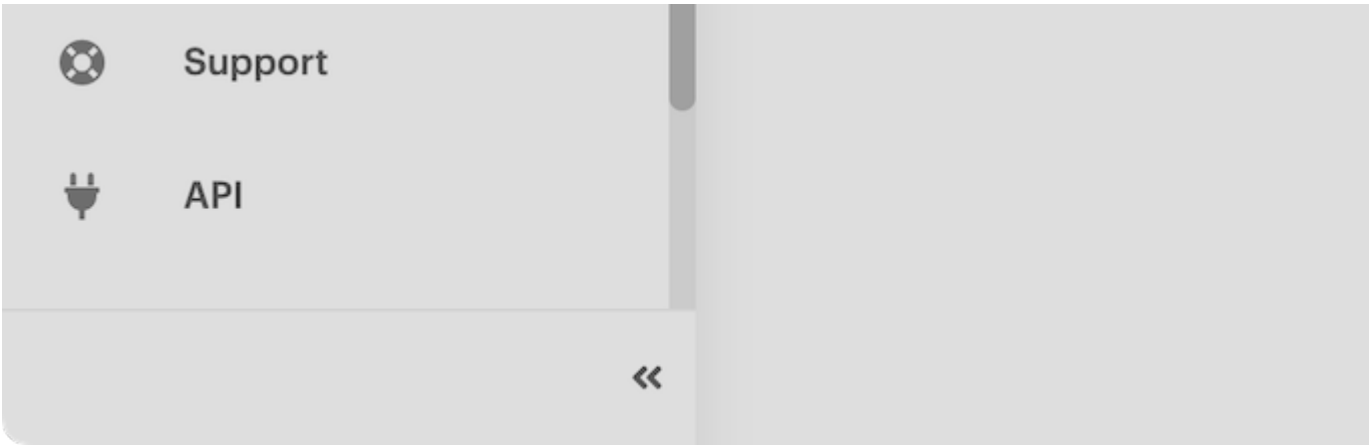
If you already have one or more public keys in your Cleura Cloud account, you can now select a ke
no public keys to choose from, activate the *Password login enabled* option and set a password for a



A configuration script is automatically prepared based on the choices you have already made. That
repositories. Click on *Advanced Options* to see the default script.


cleura
Dashboard
Compute
Networking
Storage
Images
Security Groups
Orchestration
Kubernetes
Logs
Users
Invoices
Support
API
/ Storage / Volumes
Reg
Stockholm / Sweden (0:0)
Tokyo / Japan (0:0)
Frankfurt / Germany (0:0)
Dubai / UAE (0:0)
Karlskrona / Sweden (0:0)
No resources found

It is now time to create your Cleura Cloud server; click the green *Create* button, and the new serve



An `openstack` command for creating a server may look like this:

```
openstack server create \
  --flavor $FLAVOR_NAME \
  --image $IMAGE_NAME \
  --boot-from-volume $VOL_SIZE \
  --network $NETWORK_NAME \
  --security-group $SEC_GROUP_NAME \
  --key-name $KEY_NAME \
  --wait \
  $SERVER_NAME
```

Each variable represents a piece of information we have to look for or, in the cases of `KEY_NAME` a

Let us begin with the *flavors* (`FLAVOR_NAME`), which describe combinations of CPU core count and

```
openstack flavor list
```

You will get a pretty long list of flavors. For our demonstration, we suggest you go with `b.1c1gb` . A

```
FLAVOR_NAME="b.1c1gb"
```

Your server should have an image to boot off of (`IMAGE_NAME`). For a list of all available images in

```
openstack image list
```

This time you get a shorter list, but you can still filter for images with the OS you prefer. For examp

```
openstack image list --tag "os:ubuntu"
```

Continue with the `Ubuntu 22.04 Jammy Jellyfish 20220810` image:

```
IMAGE_NAME="Ubuntu 22.04 Jammy Jellyfish 20220810"
```

Before you go on, decide on the capacity (in gibibytes) of the server's boot volume (`VOL_SIZE`). We

```
VOL_SIZE="20"
```

You need at least one network in the region you're about to create your new server (`NETWORK_NAM`

```
openstack network list --internal -c Name
```

```
+----------------+
| Name           |
+----------------+
| nordostbahnhof |
+----------------+
```

Set the `NETWORK_NAME` variable accordingly:

```
NETWORK_NAME="nordostbahnhof"
```

Regarding the security group (`SEC_GROUP_NAME`), unless you have already created one yourself, y

```
openstack security group list -c Name -c Description
```

```
+---------+-----------------------+
| Name    | Description           |
+---------+-----------------------+
| default | Default security group |
+---------+-----------------------+
```

Go ahead and set `SEC_GROUP_NAME` :

```
SEC_GROUP_NAME="default"
```

You most likely want a server you can remotely connect to via SSH without typing a password. Upl

```
openstack keypair create --public-key ~/.ssh/id_ed25519.pub bahnhof
```

In the example above, we uploaded the public key `~/.ssh/id_ed25519.pub` to our Cleura Cloud accoun

```
KEY_NAME="bahnhof"
```

By the way, check all uploaded public keys…

```
openstack keypair list
```

…and get more information regarding the one you just uploaded:

```
openstack keypair show bahnhof
```

You are almost ready to create your new server. Decide on a name…

```
SERVER_NAME="zug" # just an example
```

…and then go ahead and create it:

```
openstack server create \
    --flavor b.1c1gb \
    --image $IMAGE_NAME \
    --boot-from-volume 20 \
    --network nordostbahnhof \
    --security-group default \
    --key-name bahnhof \
    --wait \
    zug
```

(For clarity's sake, and with the exception of `IMAGE_NAME`, we used the actual values and not the v
in Cleura Cloud.

To connect to your server remotely, you need to create a floating IP for the external network in the

```
openstack floating ip create ext-net
```

See all floating IPs…

```
openstack floating ip list
```

…and assign the one you just created to your server:

```
openstack server add floating ip zug 198.51.100.12
```

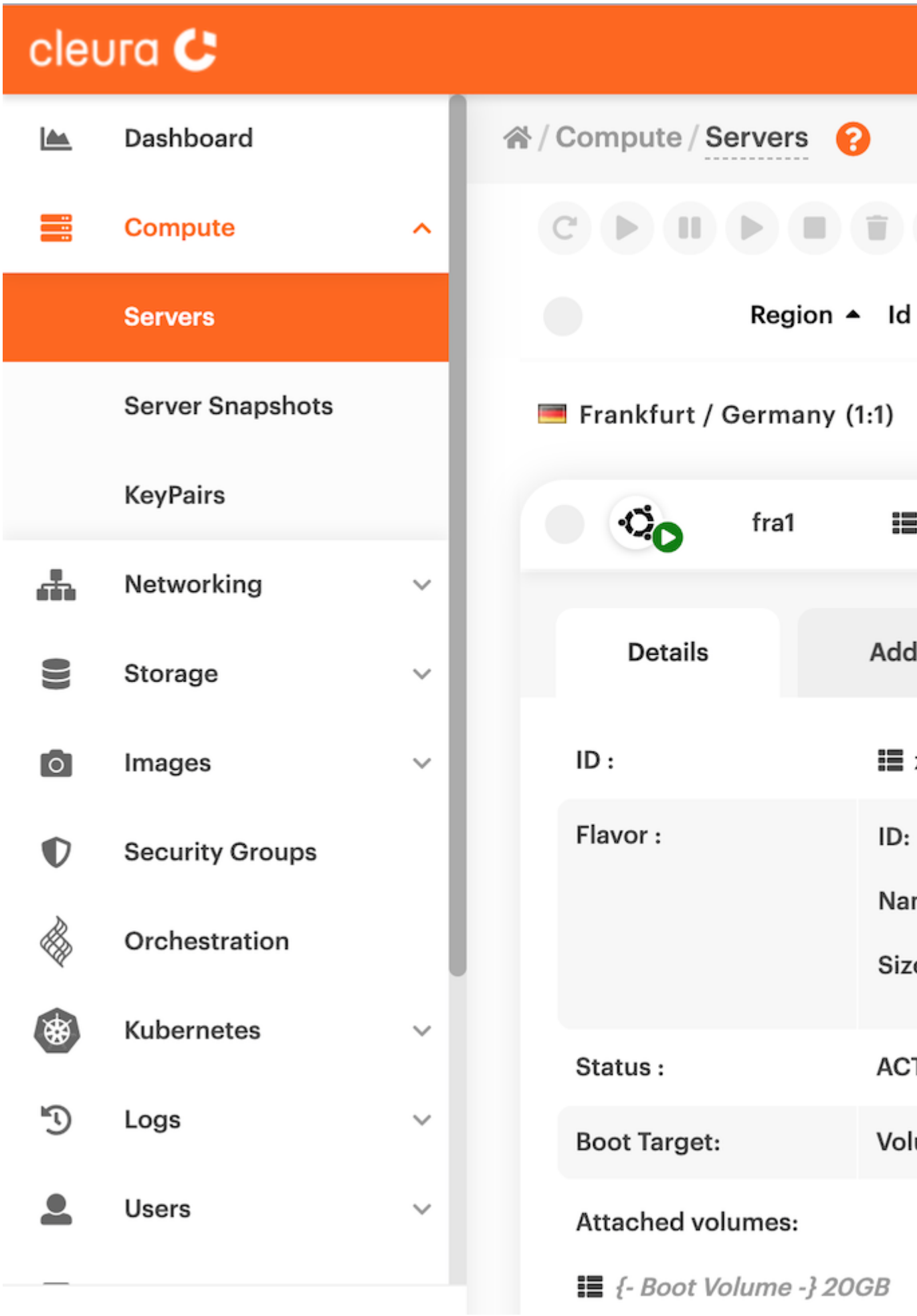
The username of the default user account in the Ubuntu image is `ubuntu`, so now you can connect t

```
ssh ubuntu@198.51.100.12
```

Viewing information about the newly created server

Cleura Cloud Management Panel **OpenStack CLI**

From the Cleura Cloud Management Panel you may, at any time, see all servers and get detailed in


cleura
Dashboard
Compute
Servers
Server Snapshots
KeyPairs
Networking
Storage
Images
Security Groups
Orchestration
Kubernetes
Logs
Users
/ Compute / Servers
Region
Id
Frankfurt / Germany (1:1)
fra1
Details
Add
ID :
Flavor :
ID:
Nar
Size
Status :
ACT
Boot Target:
Vol
Attached volumes:
{- Boot Volume -} 20GB

To see all available servers in the region, type:

```
openstack server list
```

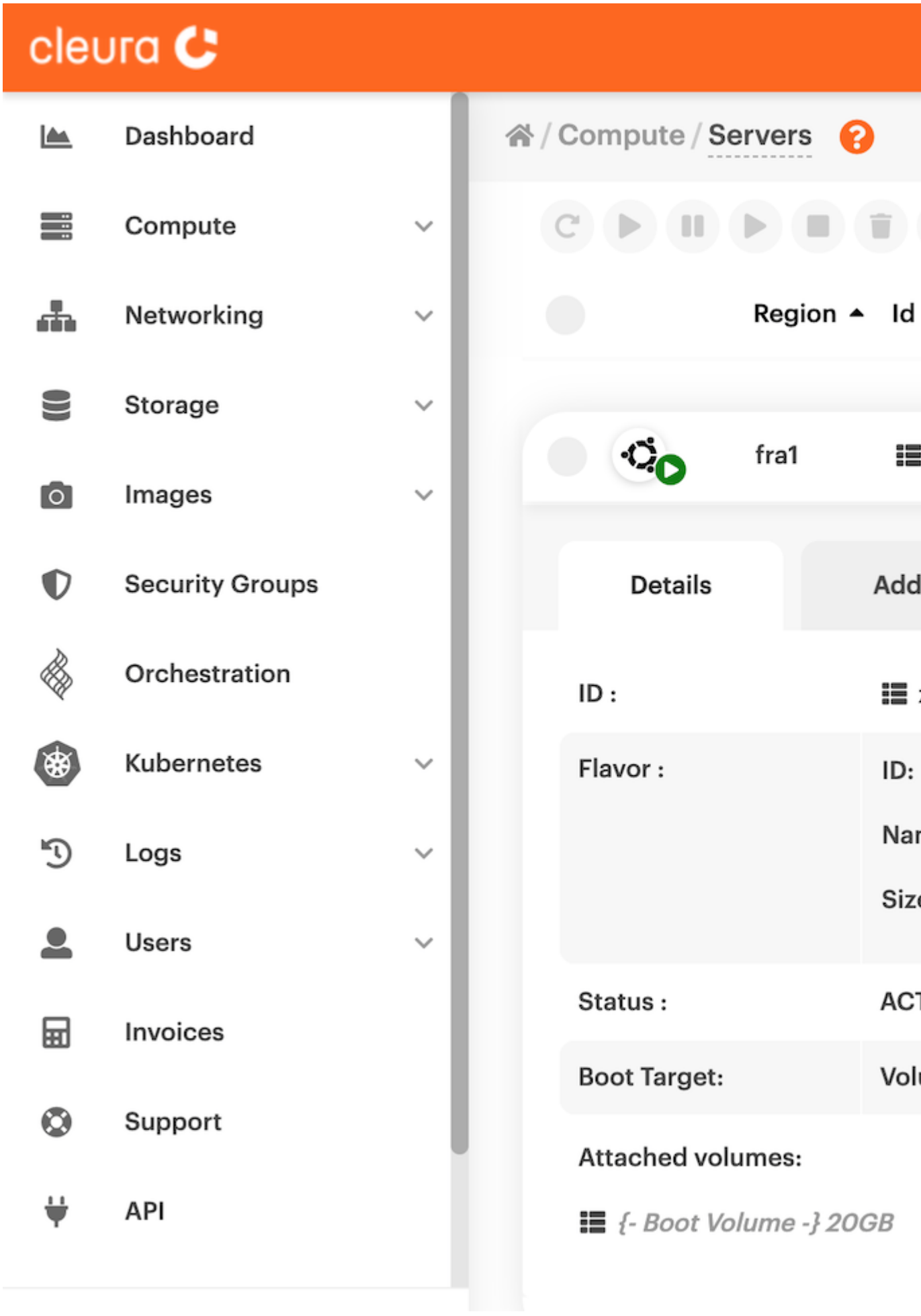
You can always get specific information on a particular server:

```
openstack server show zug
```

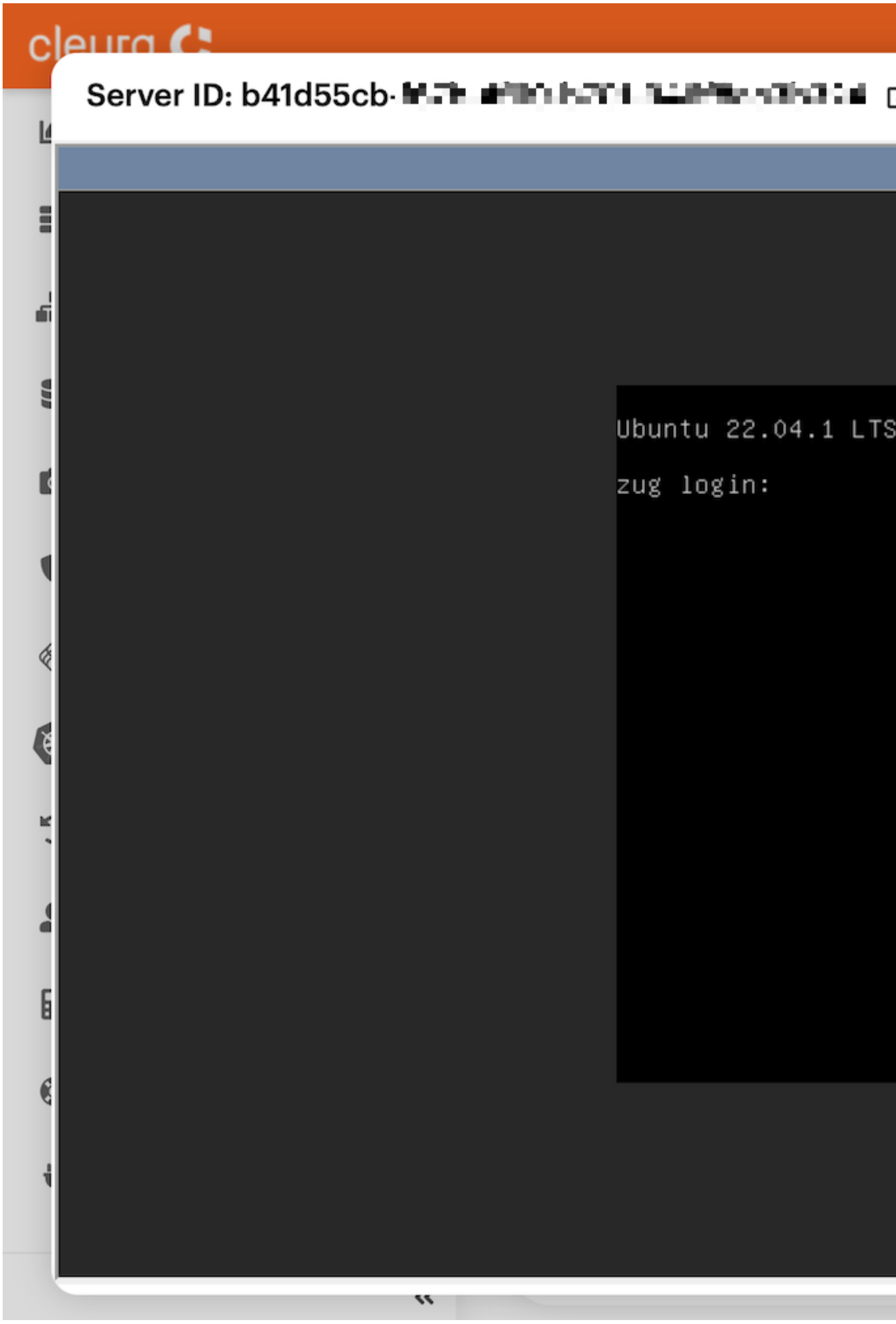
Connecting to the server console

Cleura Cloud Management Panel **OpenStack CLI**

While viewing information regarding your server, you may get its public IP address (e.g., from the A
up menu that appears select *Remote Console*.


cleura
Dashboard
Compute
Networking
Storage
Images
Security Groups
Orchestration
Kubernetes
Logs
Users
Invoices
Support
API
/ Compute / Servers
Region
Id
fra1
Details
Add
ID :
Flavor :
ID:
Nar
Siz
Status :
ACT
Boot Target:
Vol
Attached volumes:
{- Boot Volume -} 20GB

A new window pops up, and that’s your web console to your Cleura Cloud server. Please note that t


Server ID: b41d55cb-
Ubuntu 22.04.1 LTS
zug login:

You may have access to the web console of your server, and you need the corresponding URL for it

```
openstack console url show zug
```

Usage of the web console is discouraged, though. Instead, securely connect to your server via SSH

Last update: 2022-11-08
Created: 2022-11-03
Authors: Christos Varelas, Florian Haas