Universität Göttingen
Institute for Computer Science / GWDG
Julian Kunkel, Jonathan Decker

Exercise Introduction

This sheet describes the format of the course and instructions on how to connect to the GWDG compute cluster via SSH as well as how to prepare the cloud environment for the hands-on exercises.

Course Format: Practicals

The course *High-Performance Computing System Administration 2024/25* takes place in an online format as a block course from the 07.10.24 to 11.10.24 and utilizes Big Blue Button (BBB) rooms.
The first day will cover the basics of practical usage of Linux and HPC. The later days of the block course will go more in depth on topics regarding the system administration aspects of high-performance computing.
On the 11.10.24, we are offering, optional, on-site sessions in our compute center (the RzGö) at Burckhardt-Weg 4 including tours of the data center. For the on-site session, participation is limited to 20 people and a separate sign-up will be sent out.
The main BBB room is called **HPCSA**. In this room the lecturer will present the slides and guide you through the course.
As this course is intended to provide hands-on experience, the lecturers will ask you to complete exercises during the course. These exercises should be completed individually, however, you will form groups to support each other in case you get stuck. To allow for communication within said groups, each group will receive its own breakout room in BBB. The second BBB room called **HPCSA - Support** will be used for this. We will use two BBB rooms as Big Blue Button is limited, and it is currently not possible to be connected to a breakout room while also being able to listen the main room. If you need help from outside your group, feel free to ask for help in the broadcast room where the lecturer and a few helpers will be available. The format will be explained in more detail during the first session.

As you will be working with other participants of the course, you should be able to communicate with them via microphone if possible.

Examination The university students in this course will be able to choose topics related to High-Performance Computing System Administration at the end of the block course and work on a project based on said topic. For this, each student will be assigned a supervisor who is an expert on the given topic and who will guide the student. The student is expected to hand in a report by the end of the semester, which will be graded.
For further details on the examination see `https://hps.vi4io.org/teaching/autumn_term_2024/hpcsa#examination`

For the beginning of the course it is enough to join the Course room.

Course Room: `https://meet.gwdg.de/b/jul-pfo-7mr-txo`

Support Room: `https://meet.gwdg.de/b/jul-mii-pfh-shu`

Please confirm before the course that you can connect to a BBB room and your microphone is working. You may use the Support room BBB instance to test your setup.[1]

[1] If it doesn't work, please try first `https://test.bigbluebutton.org/`, then try with a different browser.

Contents

Task 1: SSH setup and Connecting to the GWDG HPC Cluster (15 min) **2**

Task 2: Prepare Cloud Environment (15 min) **3**

3 Useful Commands **22**

Task 1: SSH setup and Connecting to the GWDG HPC Cluster (15 min)

In order to follow along with the hands on exercises of this course and to complete the student projects, you need to log in to the GWDG Scientific Compute Cluster (SCC).
If you have signed up for the course[2], you should have received a confirmation that you were added to a project on the HPC project portal.

After being registered via the HPC project portal you will receive an email with the subject 'Welcome to HPC Project "HPCSA2024"'. The registration is done by us.

In that email you will find your username in the form uNNNNN, where NNNNN is a 5 digit number. This is your username for connecting to the HPC. Next you need to create an SSH key on your machine and upload it via the Academic Cloud if you haven't done so for any other project already.

First open a terminal and verify that SSH is available on your system:

Windows 10/11:

- Search for **Powershell**, right click, run as administrator
- `Get-WindowsCapability -Online|Where-Object Name -like '*SSH*'`
  If SSH client is not installed run the following command:
  `Add-WindowsCapability -Online -Name OpenSSH.Client~~~~0.0.1.0`
- Confirm that it works by running `ssh -V`

MacOS/Linux:

- Search for **Terminal** and open it
- Check ssh is provided by running the command `ssh -V`

Next generate an SSH key by running the following command
`ssh-keygen -t ed25519`

Follow the prompts provided to generate and store your key. See this documentation page for more detailed instructions `https://docs.hpc.gwdg.de/start_here/connecting/generate_ssh_key/index.html`

To upload your key, head to `https://id.academiccloud.de/security`. In the section **SSH Public Keys** press on **ADD SSH PUBLIC KEY**.
When generating your key, two files were created, one with a .pub extensions and one without any extension. On most systems you will find them in your user directory under `.ssh` . For example, with the default name it would be `id_ed25519` and `id_ed25519.pub` . The .pub file is your public key, which you need to copy into the dialog box on Academic Cloud. The file without extensions is your private key, which as the name implies, should not be shared and will later on be used to authenticate against the HPC login. See this documentation for more detailed instructions `https://docs.hpc.gwdg.de/start_here/connecting/upload_ssh_key/index.html`

[2]If you signed up late, you might not have been added to the list of course accounts. If that is the case please check your emails again and if there is no email from the project portal, contact jonathan.decker@uni-goettingen.de

Now you can login to the HPC using the following SSH commands, replace KEYNAME with the path to your SSH private key and uNNNNN with your username.

- `ssh -i KEYNAME uNNNNN@login-mdc.hpc.gwdg.de -o ProxyCommand='ssh -W %h:%p uNNNNN@login.gwdg.de -i KEYNAME'`
- Confirm the connection and enter the SSH keys passphrase **twice** if you have set any
- Confirm that running `hostname` returns *gwdu101* or *gwdu102*

This SSH command connects you as user *uNNNNN* to the SCC using the SSH key file with the same name. The proxy command is used to connect to the SCC over login.gwdg.de. This is necessary as the SCC is only reachable from inside the GÖNET. If you are already connected via the GWDG VPN or from a device inside the GÖNET, you do not need the proxy command.

For more detailed instructions or troubleshooting please see the following documentation pages `https://docs.hpc.gwdg.de/start_here/connecting/configuring_ssh/index.html` and `https://docs.hpc.gwdg.de/start_here/connecting/login_nodes_and_example_commands/index.html`.

Hints

- If you get an error stating that the permissions of your ssh key are too open, you have to limit the files permission. Type `chmod 400 hpctrainingNN` to fix the permission.
- If you get an error stating that the format of your key is invalid, try opening the key file with a text editor and make sure it starts with `-----BEGIN OPENSSH PRIVATE KEY-----` and ends with `-----END OPENSSH PRIVATE KEY-----`. You can also try copying the content into a new file. Make sure that there is an empty line at the end of the file.

Task 2: Prepare Cloud Environment (15 min)

During the HPCSA block course you will be completing exercises that involve the installation and configuration of software under a Linux system. As this typically requires root permission on a given system, which cannot be granted on the SCC system, you will instead use Virtual Machines (VMs) handled by the GWDG OpenStack instance. The following steps will show you how to set up VMs using the OpenStack web interface and how to connect to them. During the course you will be responsible for managing your VMs.

Login

This section shows how to login into the OpenStack Dashboard.

1. Find your username and password in the course email.
2. Open `https://cloud.gwdg.de` in your browser and select **Login via AcademicID**.
3. Use your username and password to login on the AcademicID web page.
4. You should be directed back to `https://cloud.gwdg.de` and see the OpenStack Horizon Dashboard similar to Figure 1

Create SSH Key

This section shows how to create an SSH key pair, which can later be used to connect to your VMs.

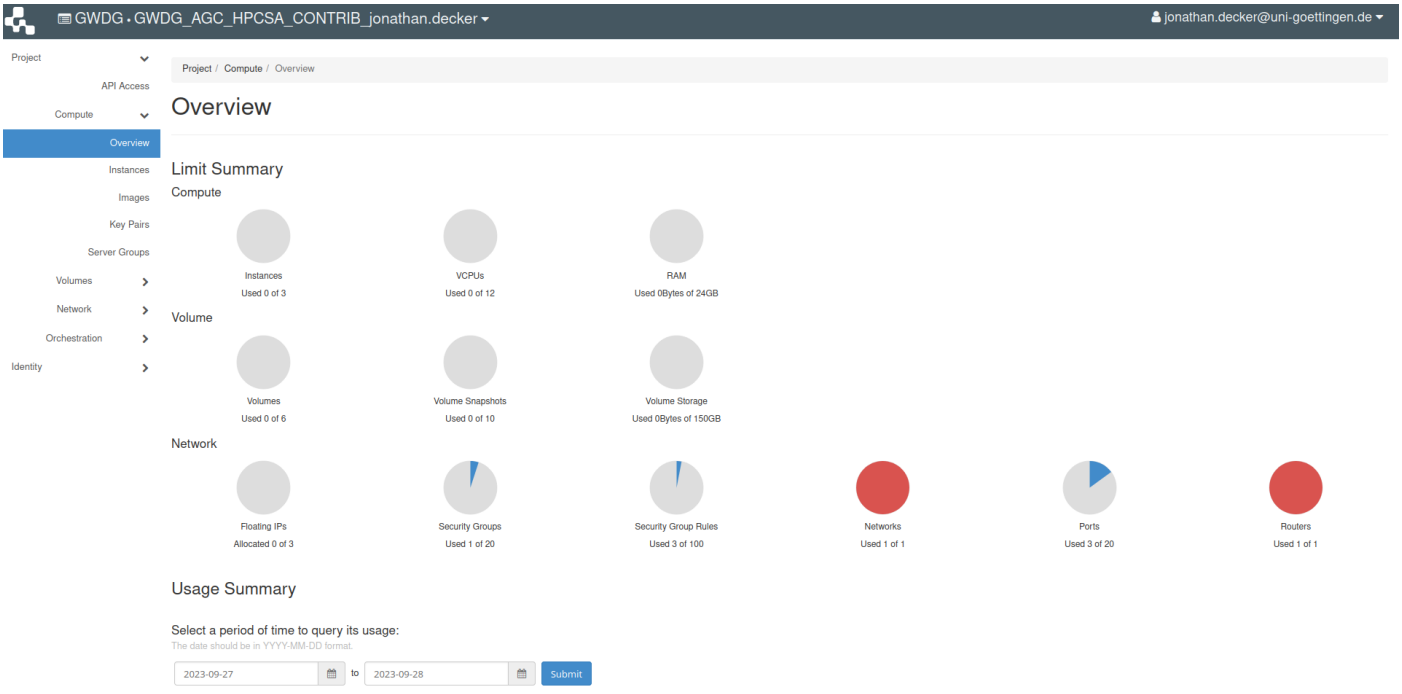

Figure 1: GWDG OpenStack Horizon Dashboard

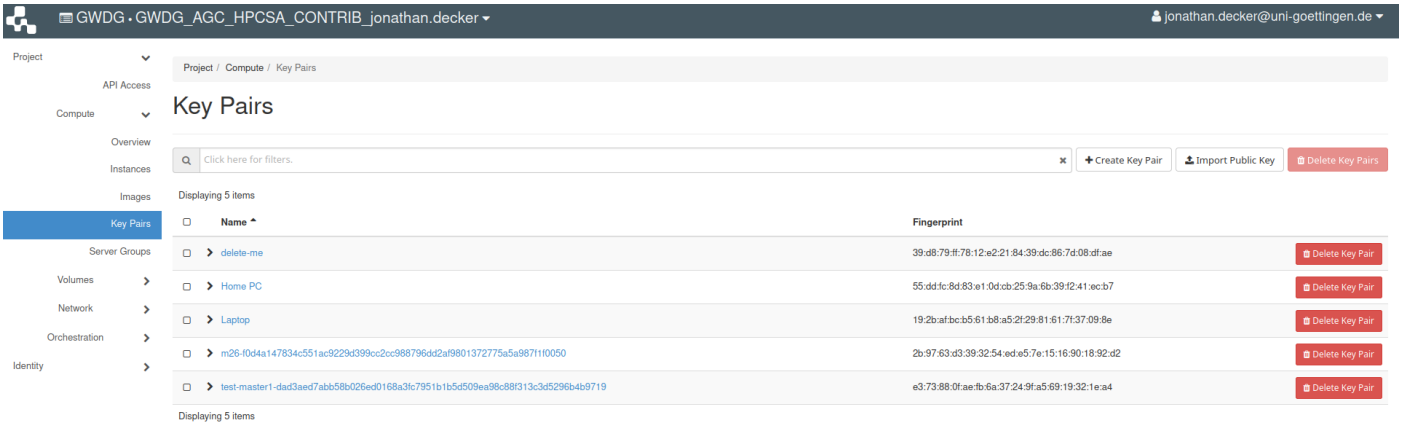

Figure 2: OpenStack key pairs

1. On the left side under **Compute** tab, select **Key Pairs**.
   You should see an empty key list similar to Figure 2 If the list is not empty and a key with the name **hpcsa-course-vm-key** already exists, delete it.

2. Click on **Create Key Pair** and name it **hpcsa-course-vm-key**. See Figure 3 for reference. Then create the key pair. This will automatically download the private key to your computer, which will be required later to connect to the VMs. You should see the created key pair in the overview.

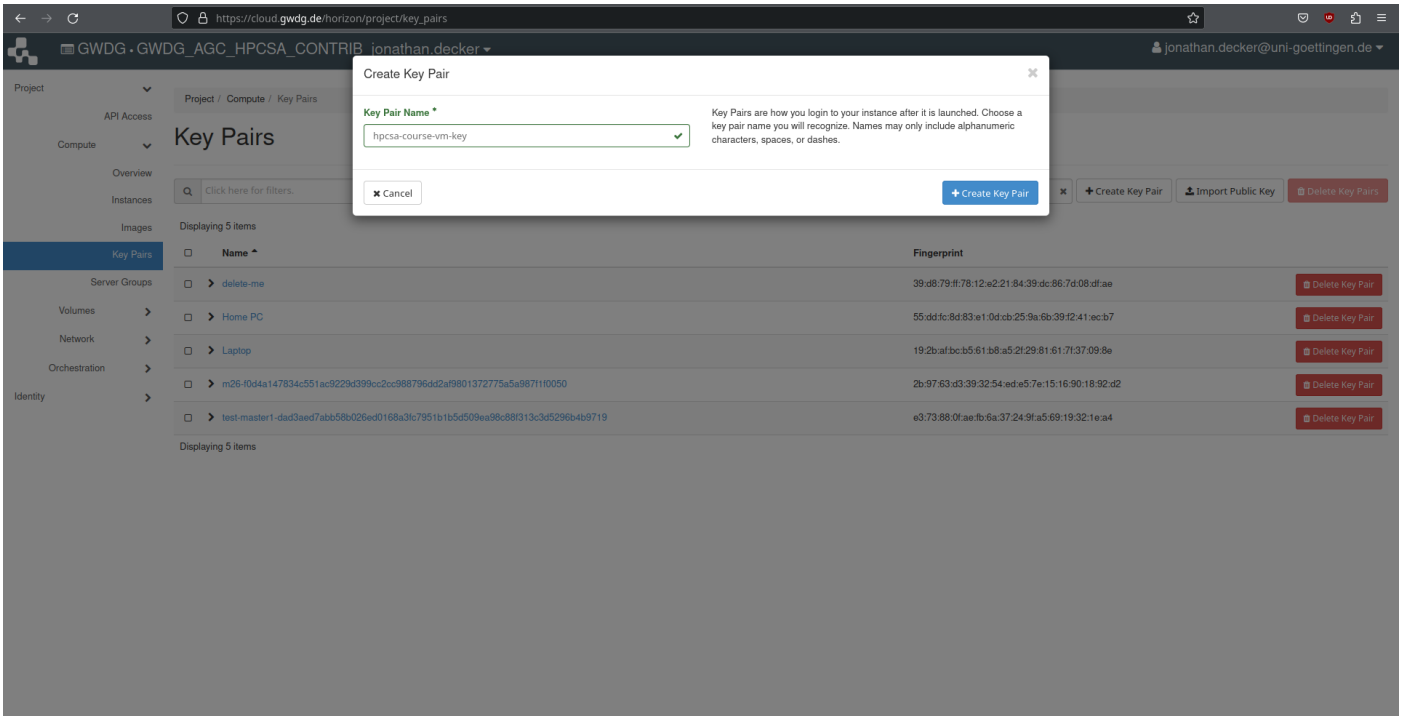

Figure 3: Create a new key pair

Configure Security Groups

This section shows how to configure the security groups on OpenStack to open ports for external and internal communication.

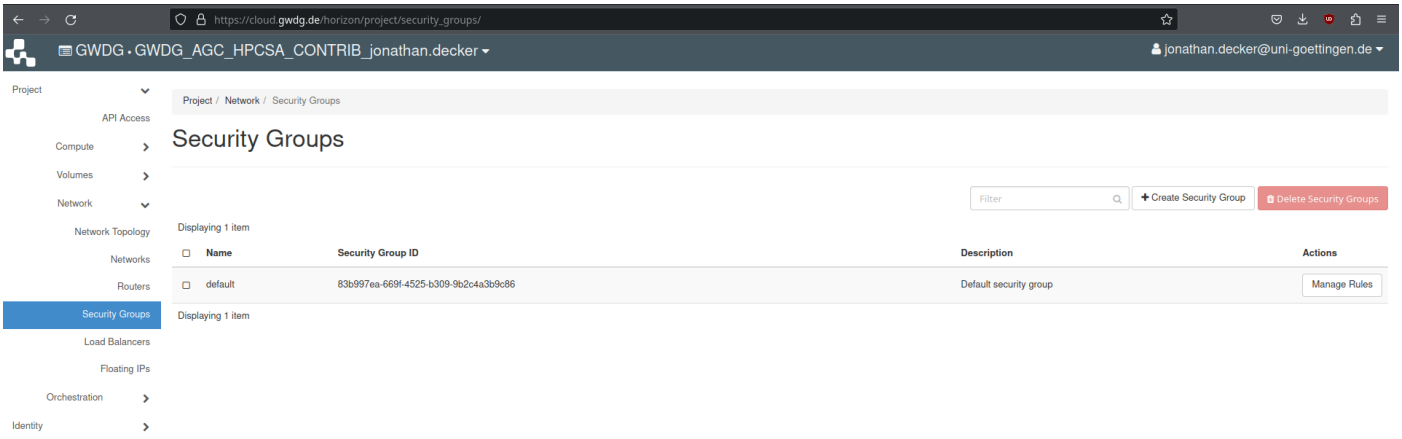

Figure 4: OpenStack Security Groups Overview

1. One the left side, under **Network** tab, select **Security Groups**. The overview should show the default

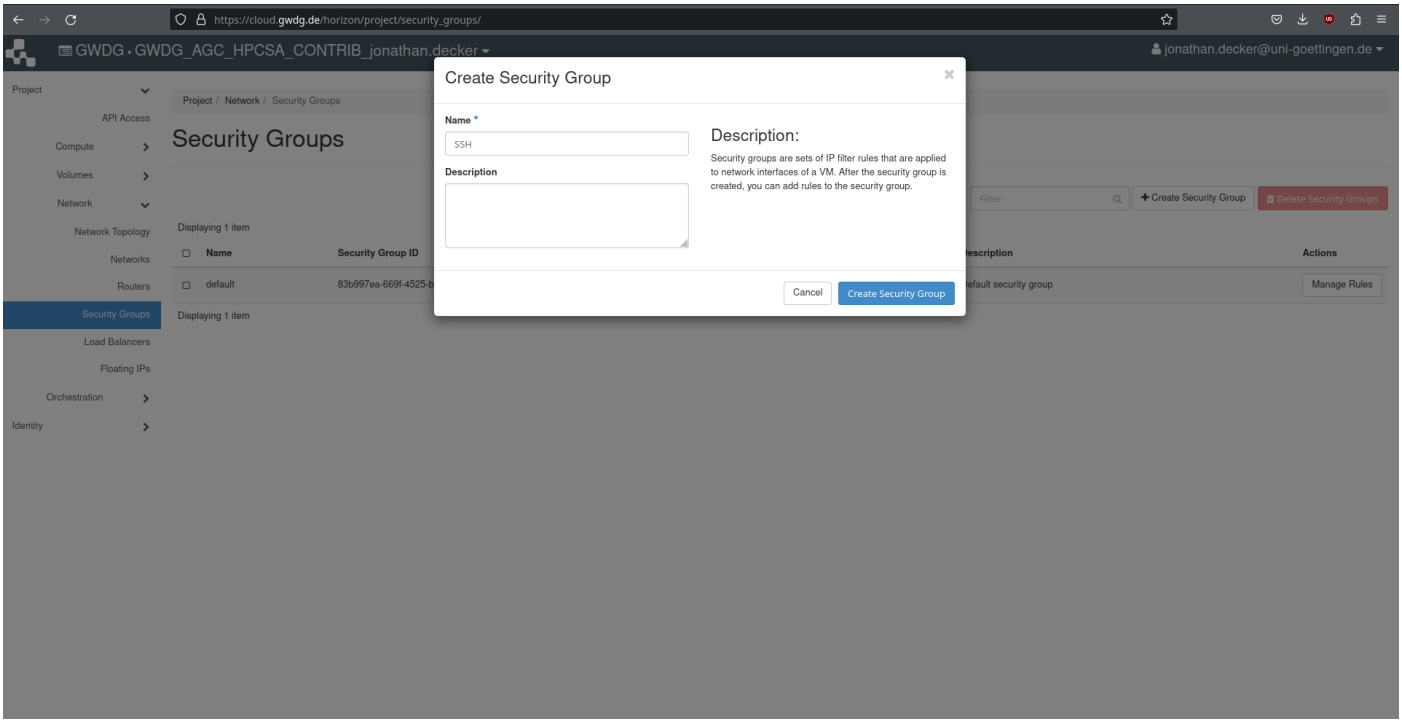

Figure 5: Security Group Creation Dialog for SSH

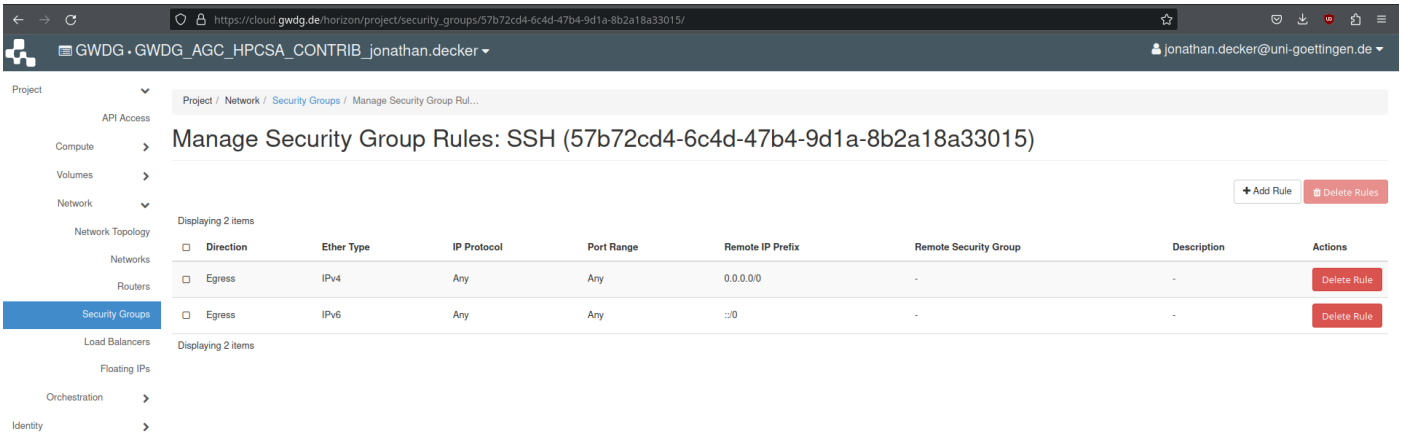

Figure 6: Manage Security Group Rules Overview

security group as shown in Figure 4.

2. If the list already contains security groups, delete them except for the default group and create new ones as described in the next steps.

3. Press **Create Security Group** and name it **SSH** as shown in Figure 5.

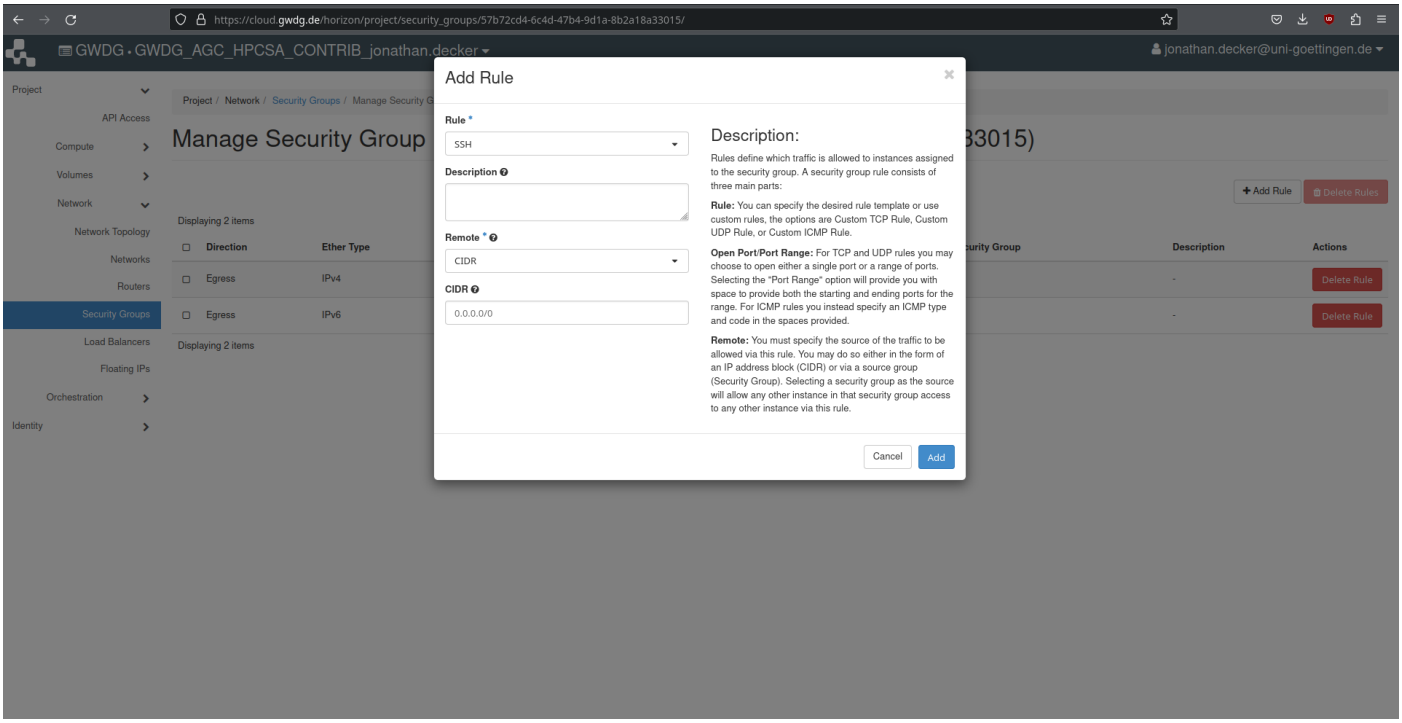

Figure 7: Add Security Group Rule Dialog for SSH

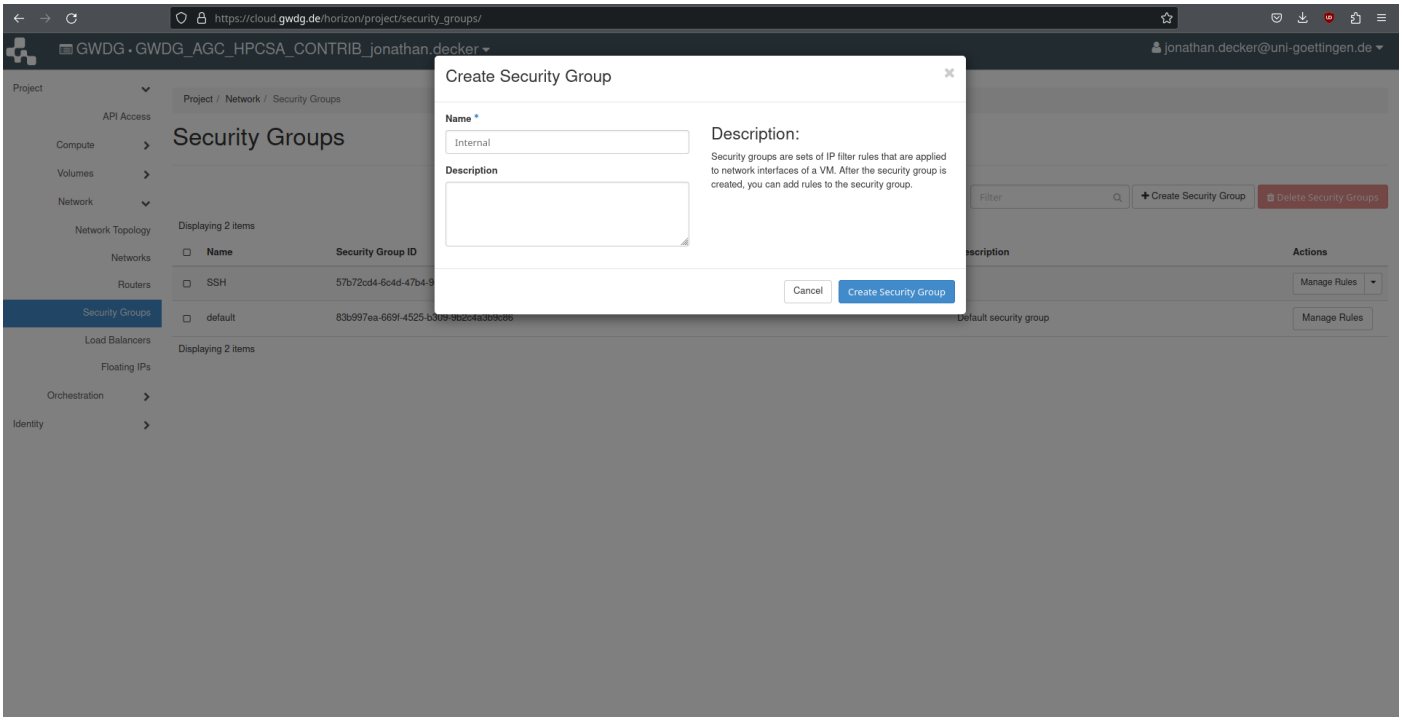

Figure 8: Security Group Creation Dialog for Internal

4. To edit the new group press **Manage Rules** and you should an overview similar to Figure 6.
5. Press **Add Rule** and in the new dialog under **Rule** select **SSH** from the drop-down menu as shown in Figure 7.
6. Save this rule, go back to the security groups overview and create another security group called **Internal** as shown in Figure 8.

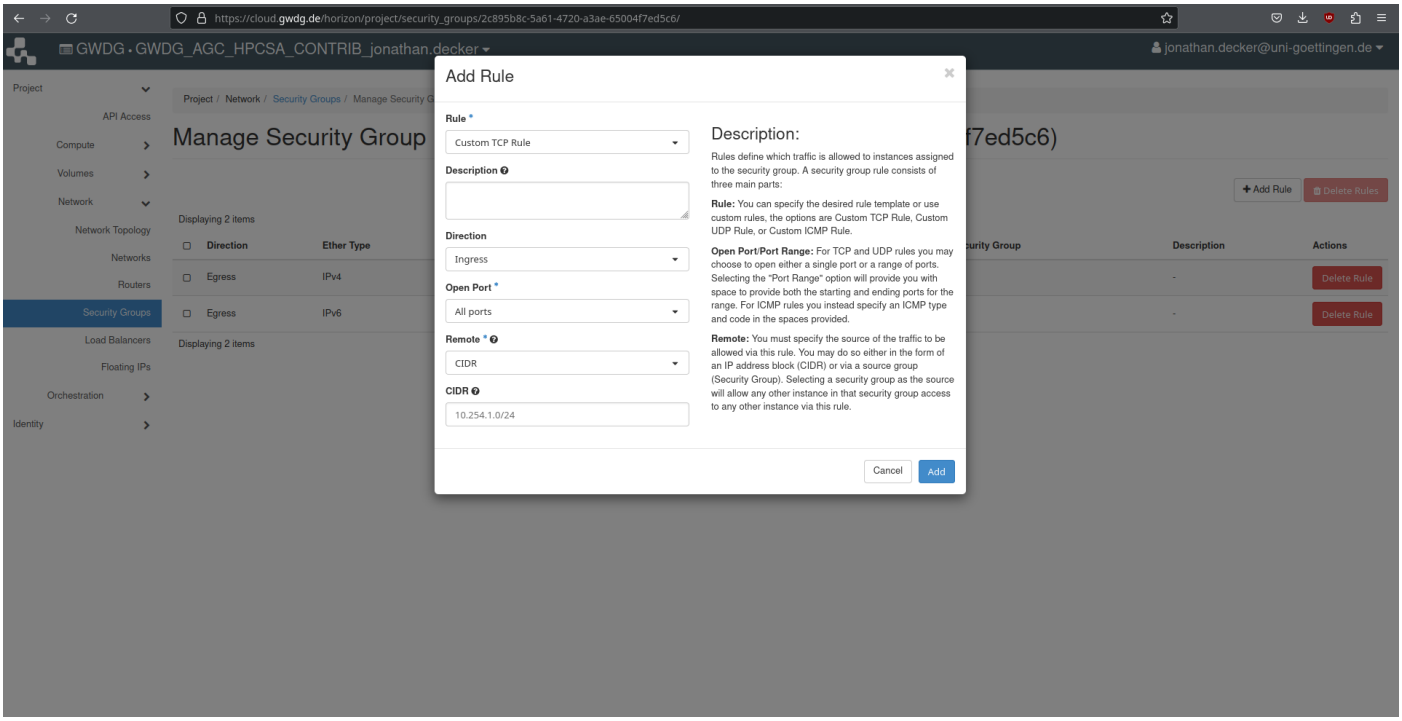

Figure 9: Add Security Group Rule Dialog for internal TCP

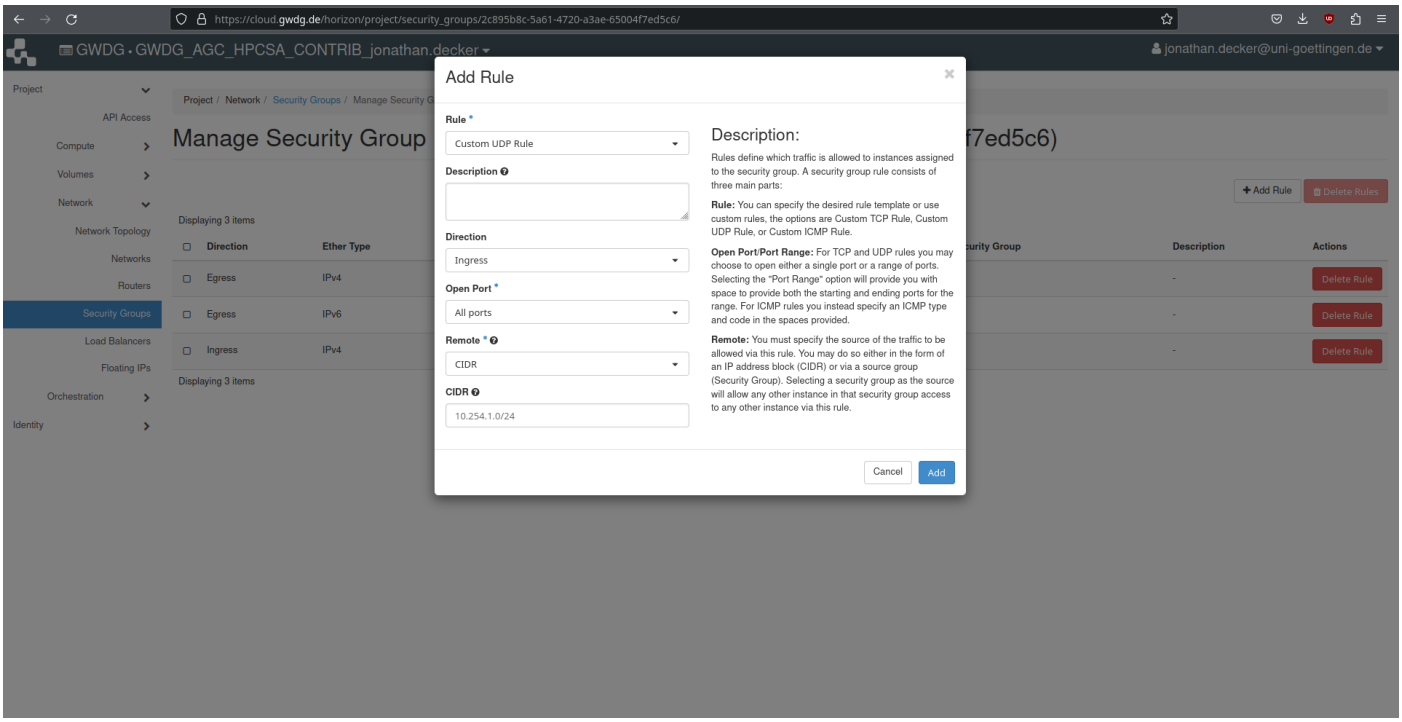

Figure 10: Add Security Group Rule Dialog for internal UDP

7. Manage the group **Internal** and add two new rules. For the first rule set **Rule** to **Custom TCP Rule** under **Open Port** set **All ports** and most importantly under **CIDR** set it to **10.254.1.0/24** as can be seen in Figure 9.
   The **CIDR** setting configures the IP mask for which this rule is valid. This rule allows incoming TCP traffic on all ports for this VM as long as it comes from an IP within the mask **10.254.1.0/24**. This mask represents the internal IP addresses used by OpenStack so only your other VMs can use this rule.

8. Create a second rule in the **Internal** group using the same settings as for the other rule but setting **Rule** to **Custom UDP Rule** as shown in Figure 10.

Launch Main Instance

This section shows how to launch your main instance, which will use Rocky 8 and be reachable from your machine via SSH.

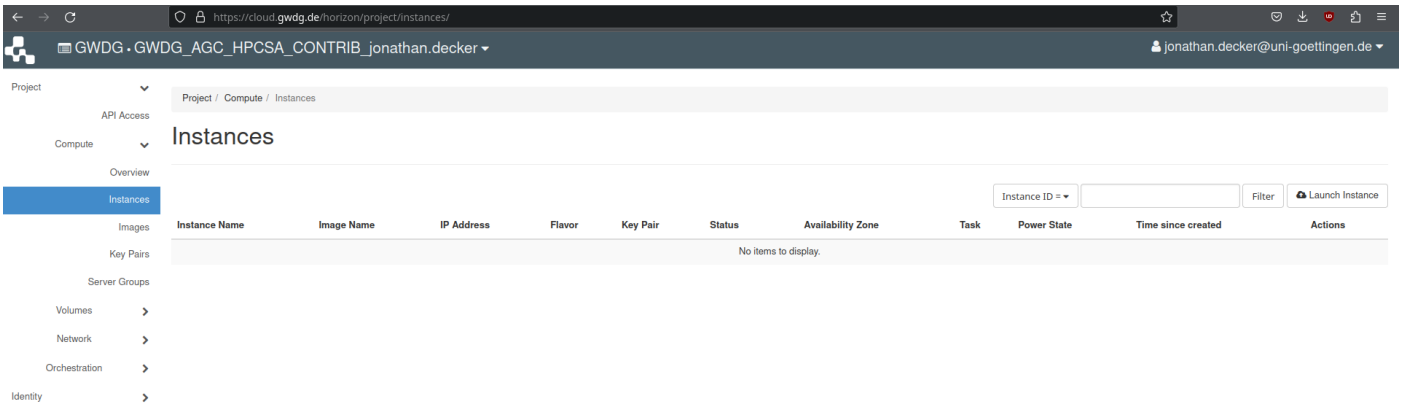

Figure 11: OpenStack Instance Overview Empty

1. On the left side, under **Compute** tab, select **Instances**. This opens the instance overview, which should show no instances as in Figure 11.
2. Click on **Launch Instance** and name your instance **cluster-manager** as shown in Figure 12.
3. Press **Next** to configure the **Source**. In the list of at the bottom find **Rocky 8** and press the up arrow on the right to select it. Then ensure that **Create New Volume** and **Delete Volume on Instance Delete** are both set to **Yes**. The dialog should look the same as shown in Figure 13.
4. Press **Next** to configure **Flavor**, which refers to the preset of compute resources your new instance will have. From the list find **m1.large** and press the arrow up on the right to select it such that it looks as shown in Figure 14.
5. Skip **Networks** and **Network Ports** and select **Security Groups**. Move both of your groups, **SSH** and **Internal** up via the arrow on the right such that it looks the same as in Figure 15.
6. Move on to **Key Pair** and select the key pair called **hpcsa-course-vm-key**, which you had created earlier via the arrow on the right such that the key is under **Allocated** and it looks similar to Figure 16.
7. Press **Launch Instance** and see that the system is now working on provisioning your VM as can be seen in Figure 17.
8. After a short time, the instance should be ready and visible in the overview, similar to Figure 18.

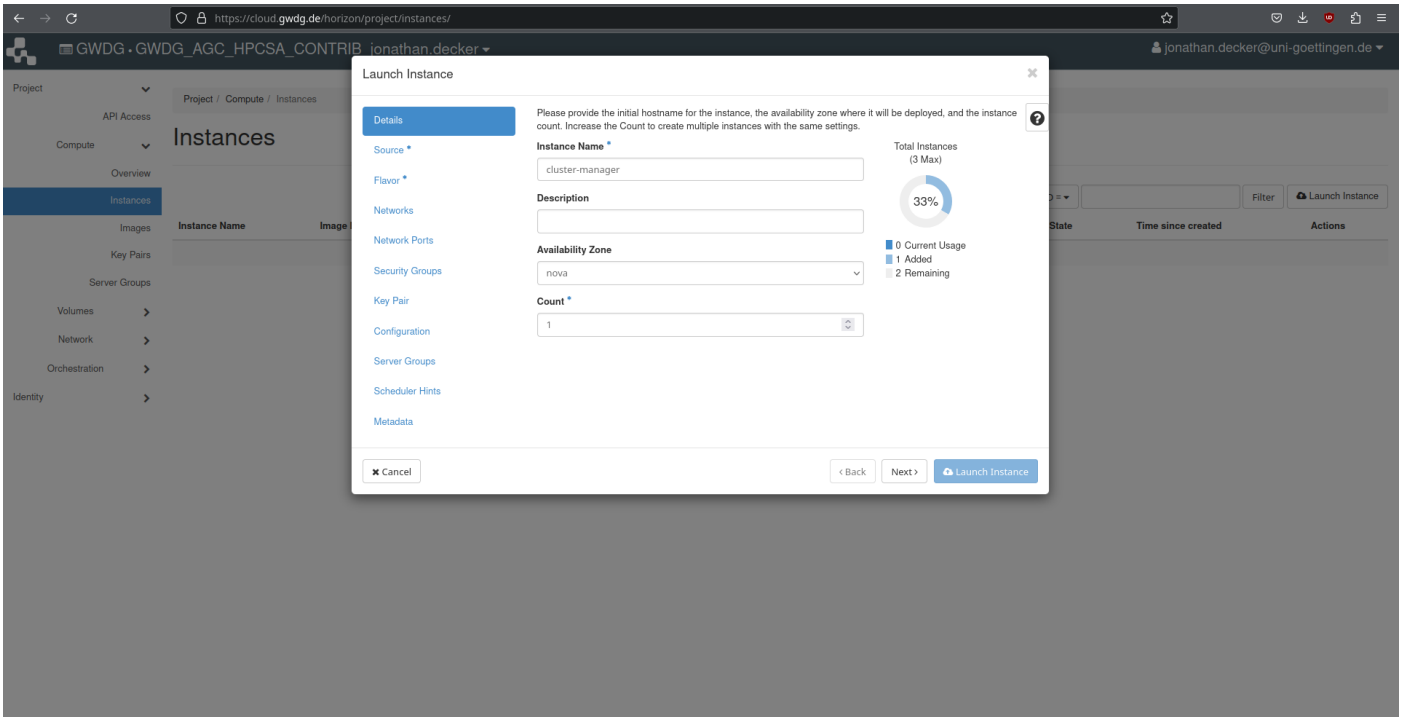

Figure 12: Launch Instance Details for Cluster-Manager

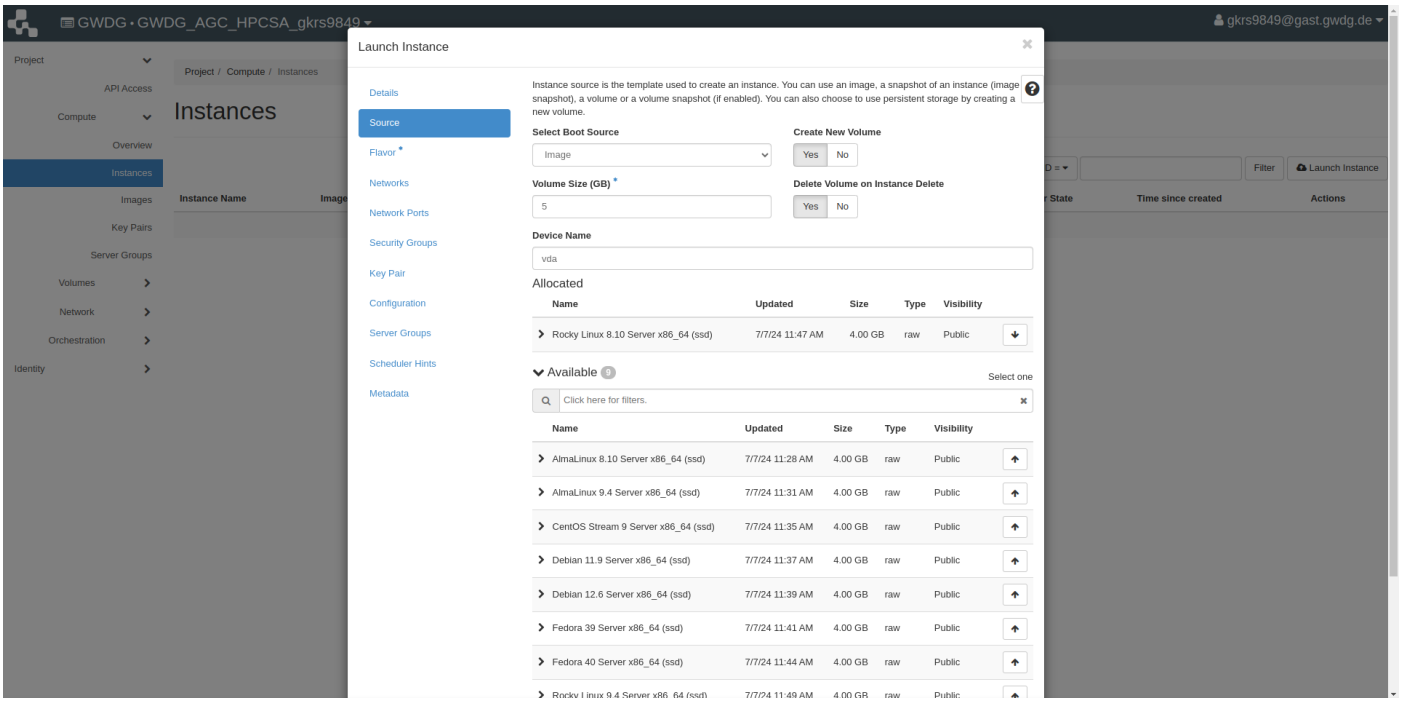

Figure 13: Launch Instance Source for Cluster-Manager

Add a Floating IP Address

This section shows how to associate a floating IP address to your VM, which makes it possible to connect to the VM.

1. Under the **Compute** tab, on **Instances**, open the drop-down menu for your **cluster-manager** instance and select **Associate Floating IP** from it as shown in Figure 19.

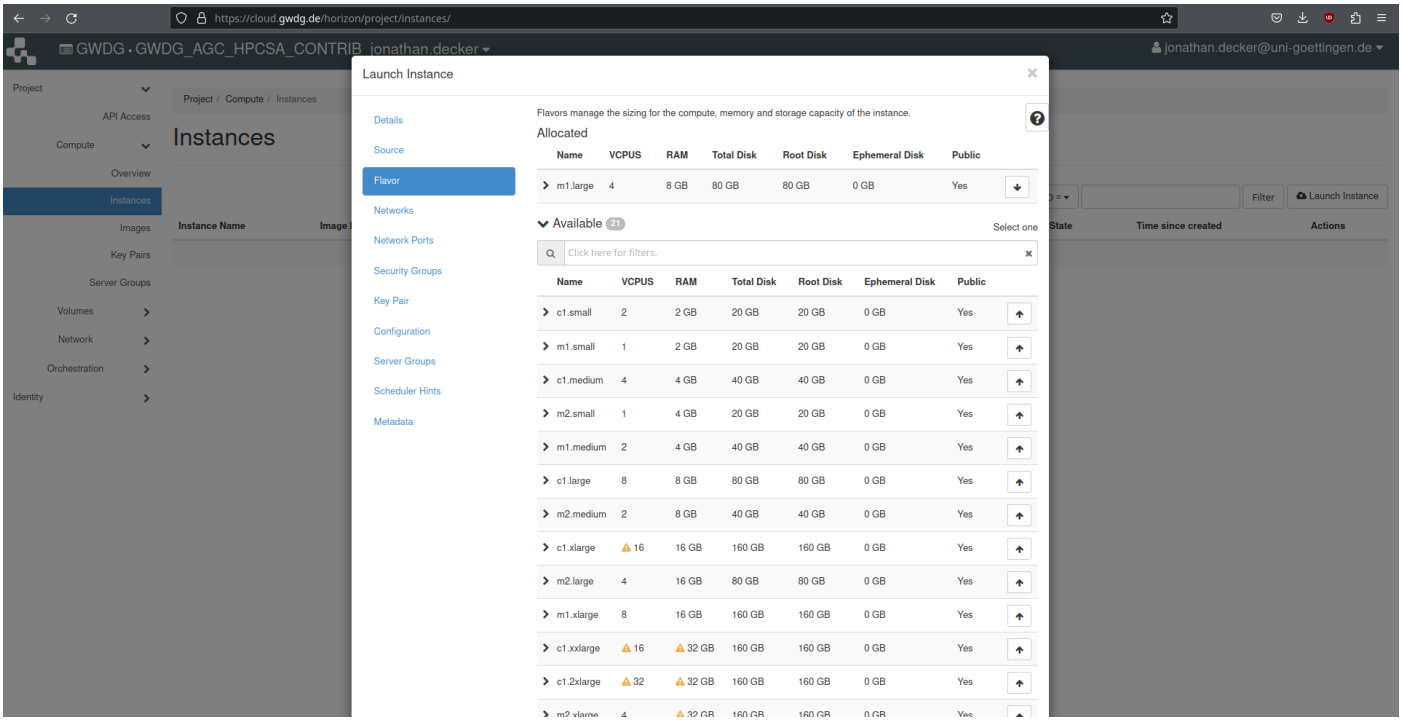

Figure 14: Launch Instance Flavor for Cluster-Manager

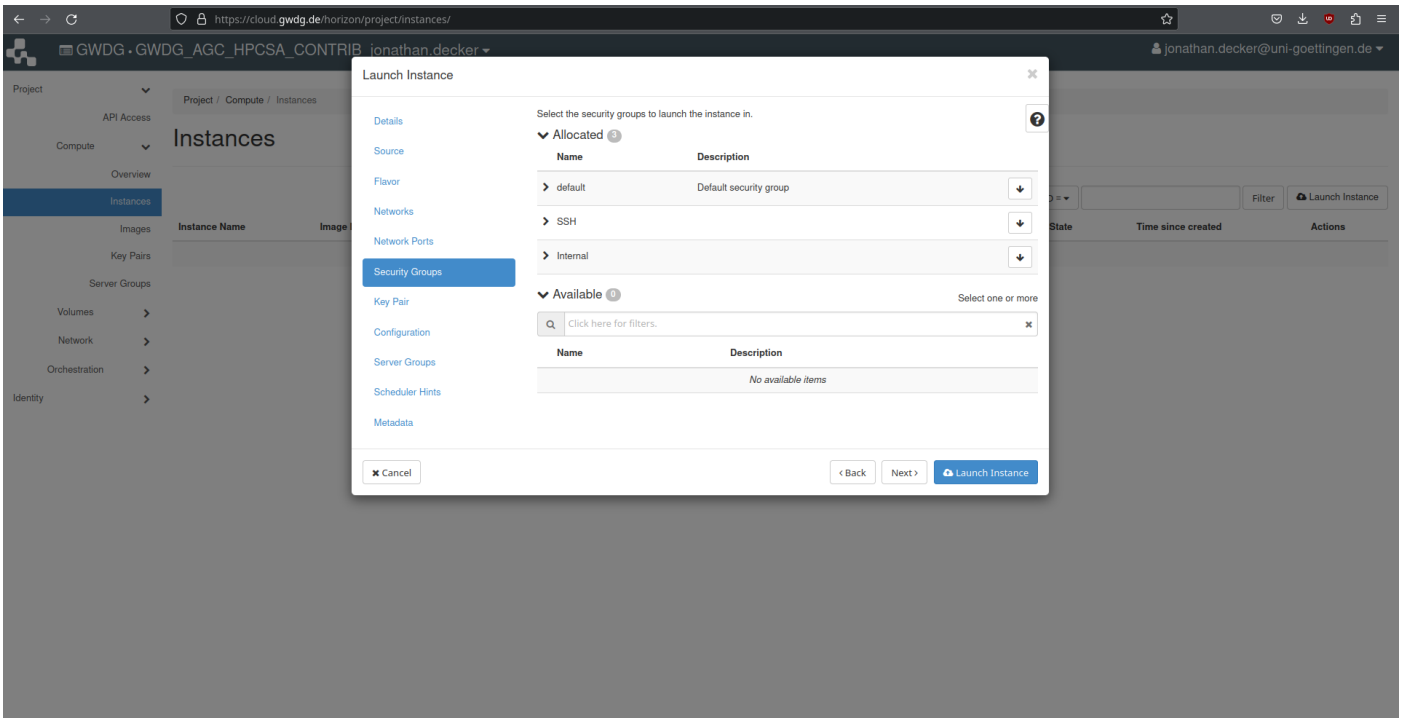

Figure 15: Launch Instance Security Groups for Cluster-Manager

2. In the new dialog, click on the plus sign next to the **IP Address** drop-down menu, as shown in Figure 20, to allocate a new IP address.
3. In the following dialog, press **Allocate IP** without changing anything as shown in Figure 21.
4. You are returned to the previous dialog, where an IPv4 address is now shown under the **IP Address** drop-down menu as shown in Figure 22. Confirm the association by pressing **Associate**.

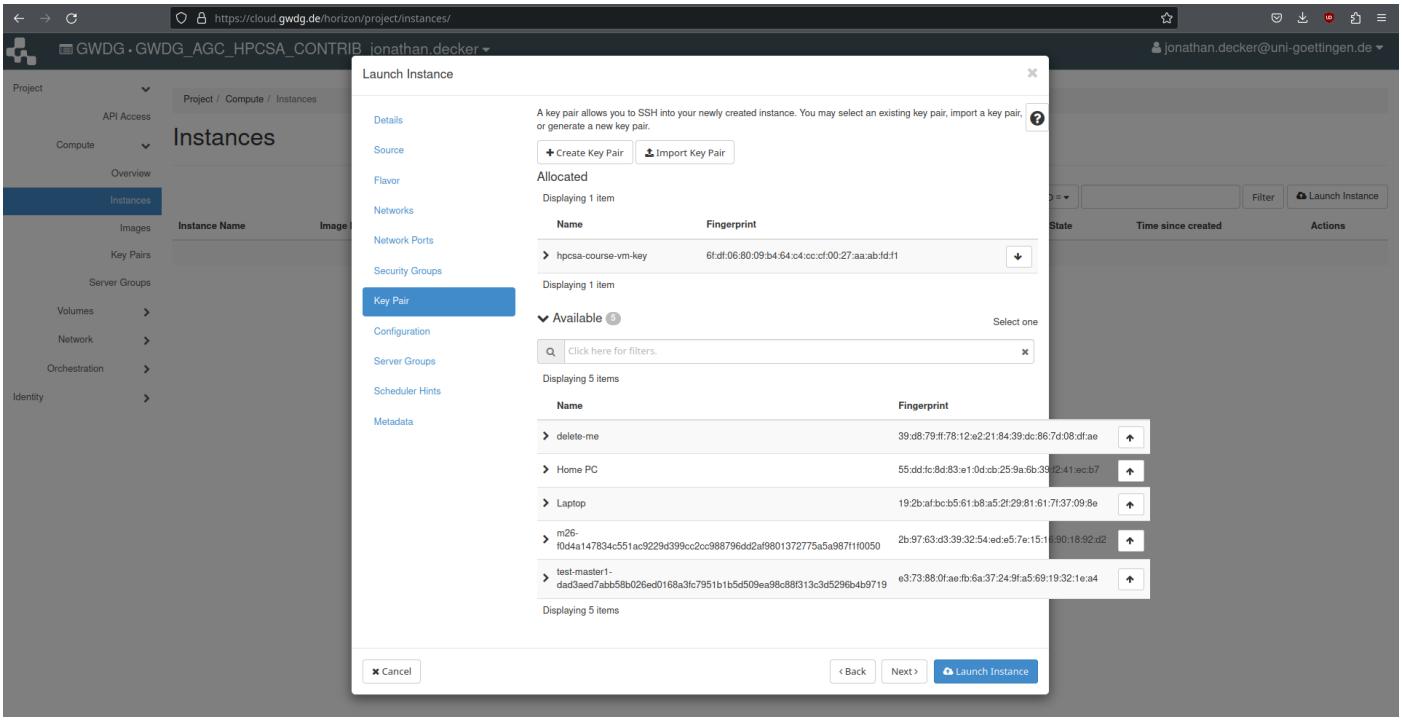

Figure 16: Launch Instance Key Pair for Cluster-Manager

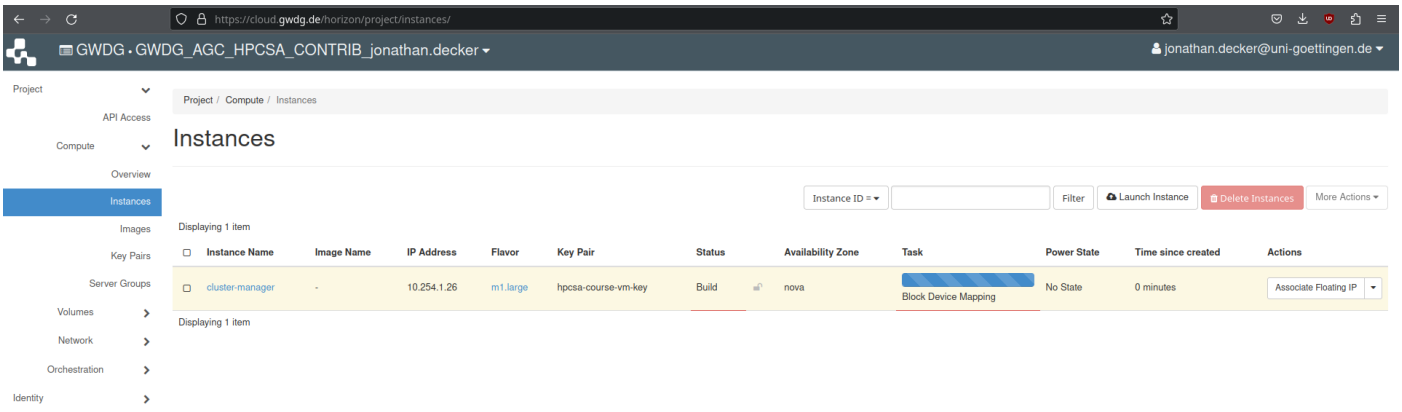

Figure 17: Cluster-Manager Instance is Launching

5. In the instances overview you should now see two IP addresses for your **cluster-manager** instance in the respective column as shown in Figure 23. Make a note of the IP noted after **Floating IPs** as you will need it in the next step.

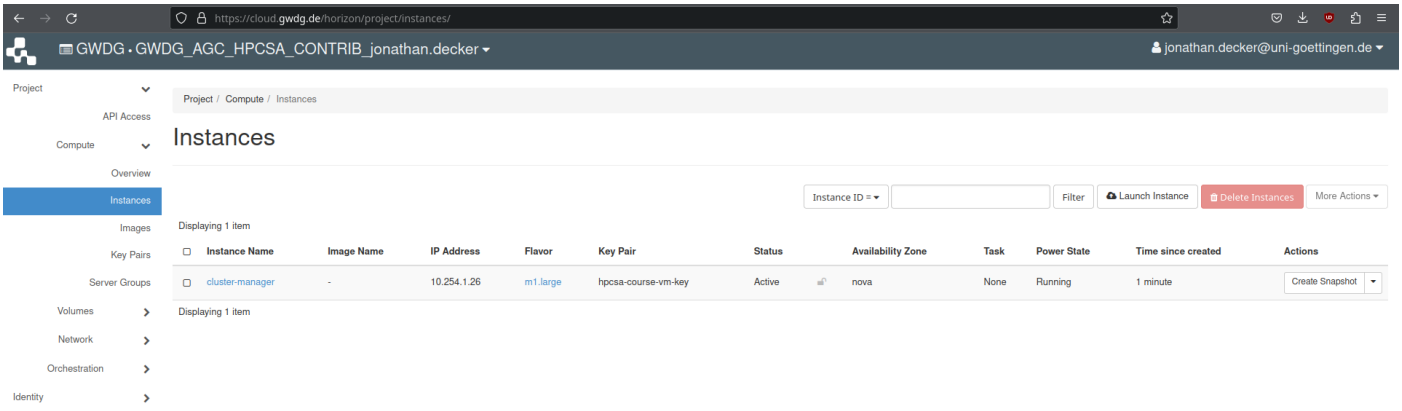

Figure 18: Cluster-Manager Instance is Running

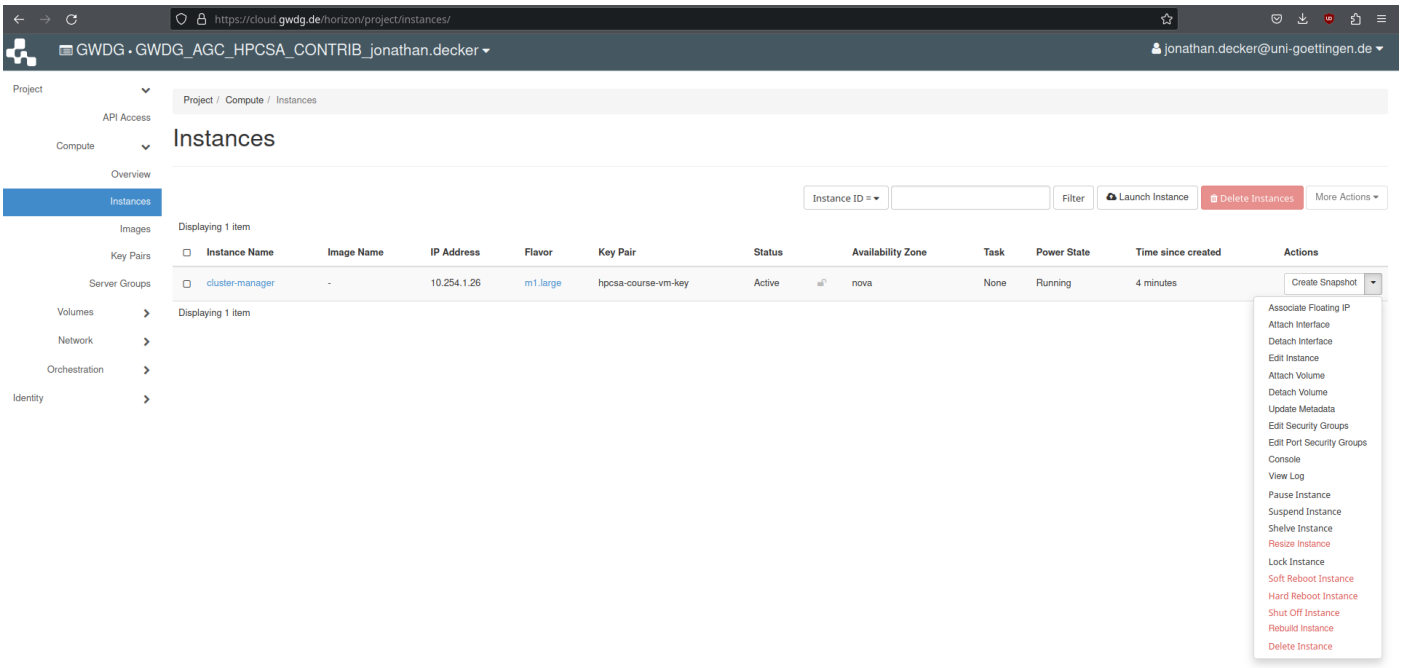

Figure 19: Cluster-Manager Actions Drop-Down Menu

Connect to Instance with SSH

This section shows how to use SSH and your SSH key to connect to your VM.

1. Find the **hpcsa-course-vm-key.pem** you downloaded in your **Downloads** folder or where you have saved it and move it into your user folder.

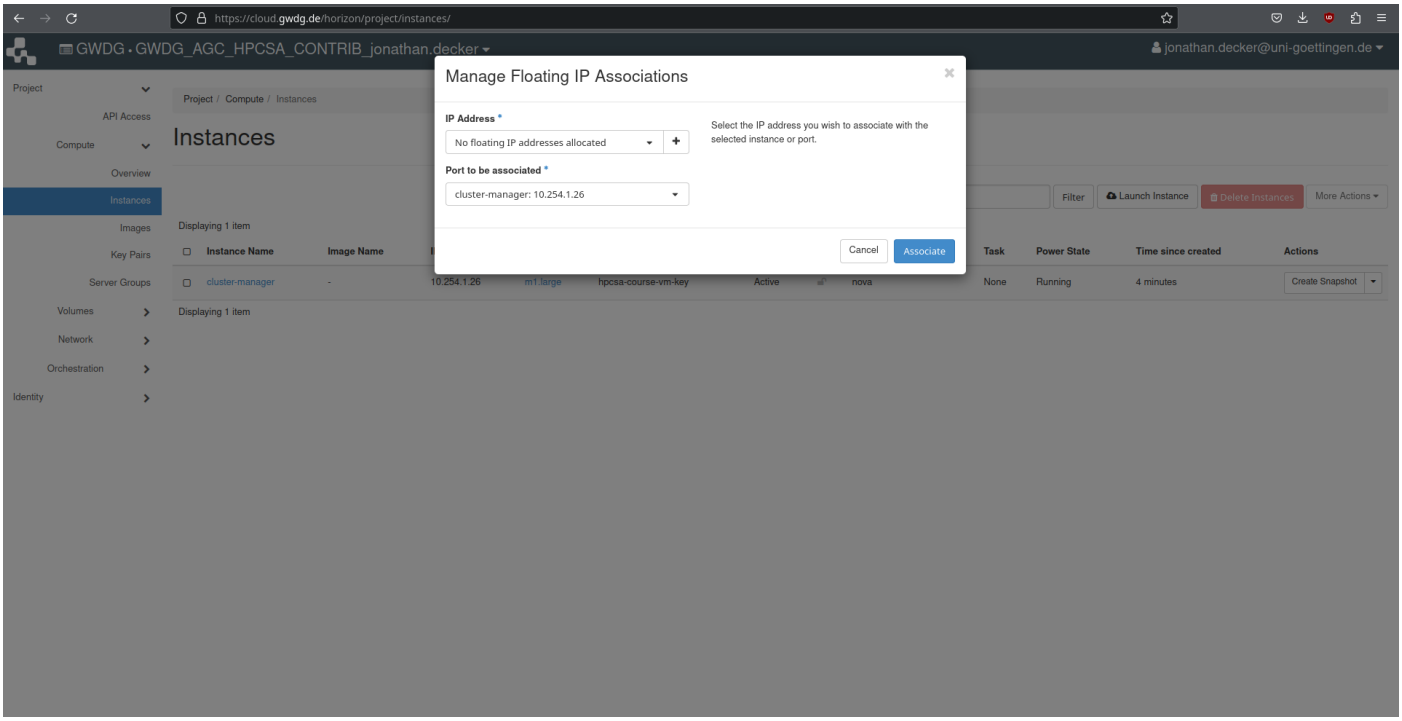

Figure 20: Manage Floating IP Dialog without Allocated Floating IP

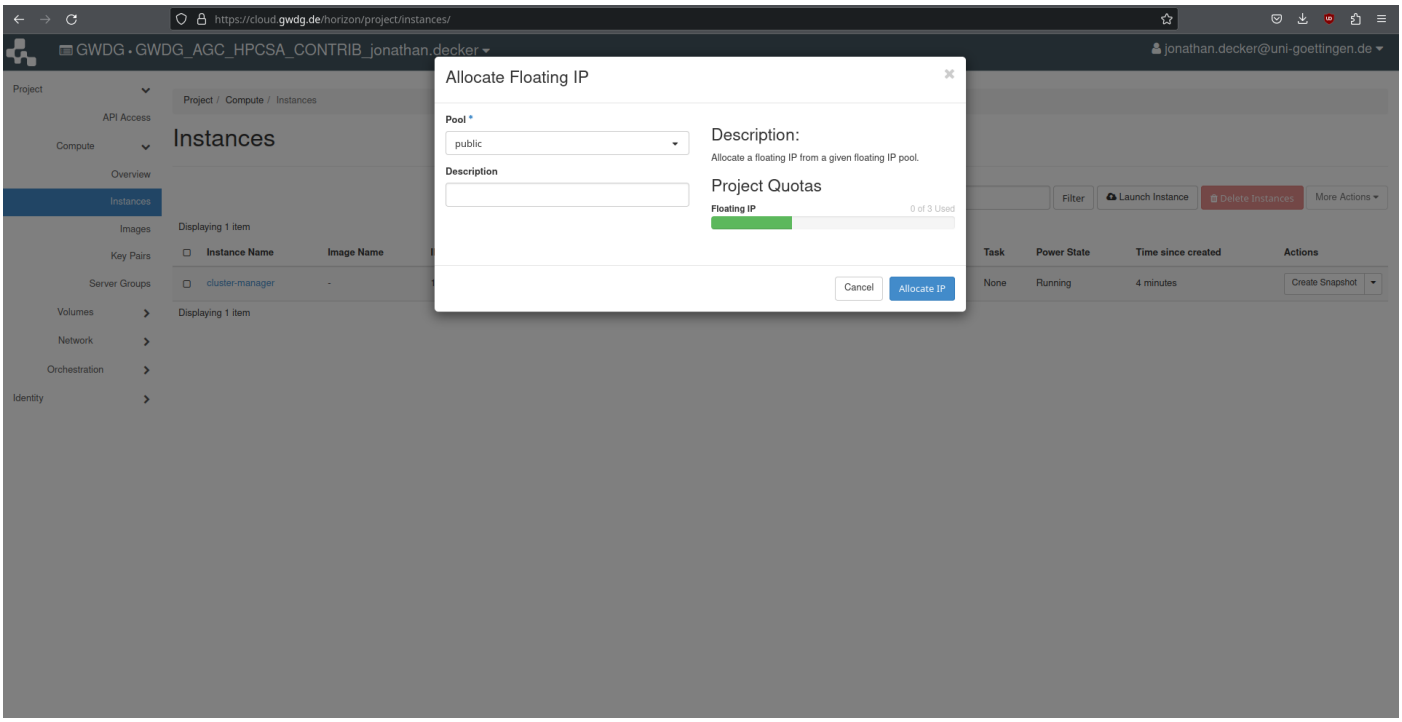

Figure 21: Allocate Floating IP Dialog

2. Open a terminal and confirm that you have **SSH** installed by following the platform specific instructions:

Windows 10/11

1. Search for **Powershell**, right click, run as administrator
2. `Get-WindowsCapability -Online|Where-Object Name -like '*SSH*'`

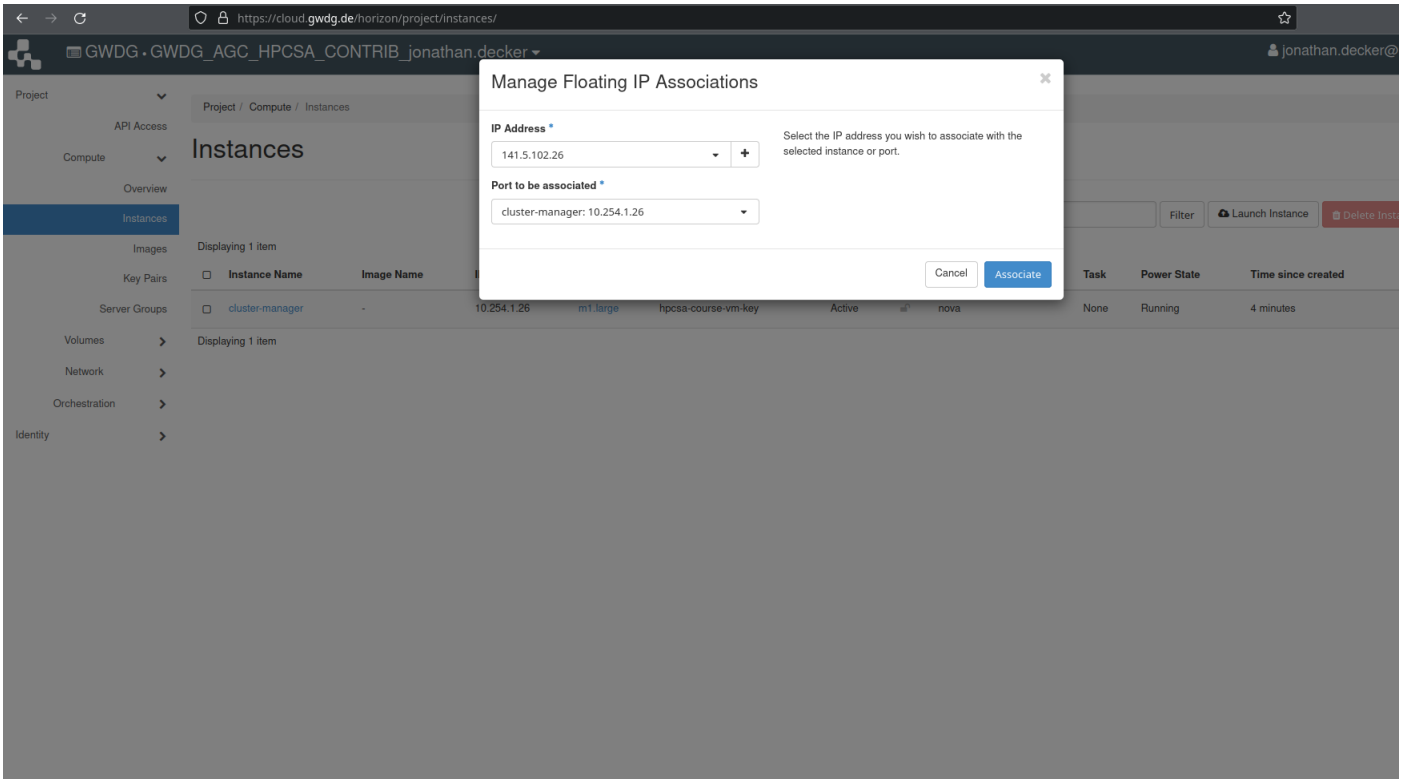

Figure 22: Manage Floating IP Dialog with Allocated Floating IP

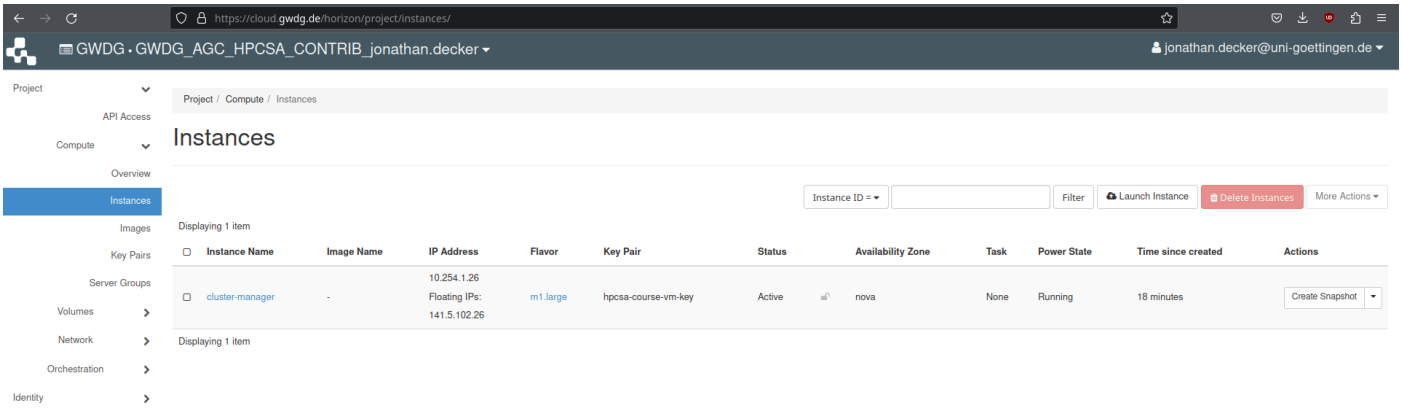

Figure 23: Instance Overview with Cluster-Manager with Floating IP

If SSH client is not installed run the following command:
`Add-WindowsCapability -Online -Name OpenSSH.Client~~~~0.0.1.0`

3. Confirm that it works by running `ssh -V`

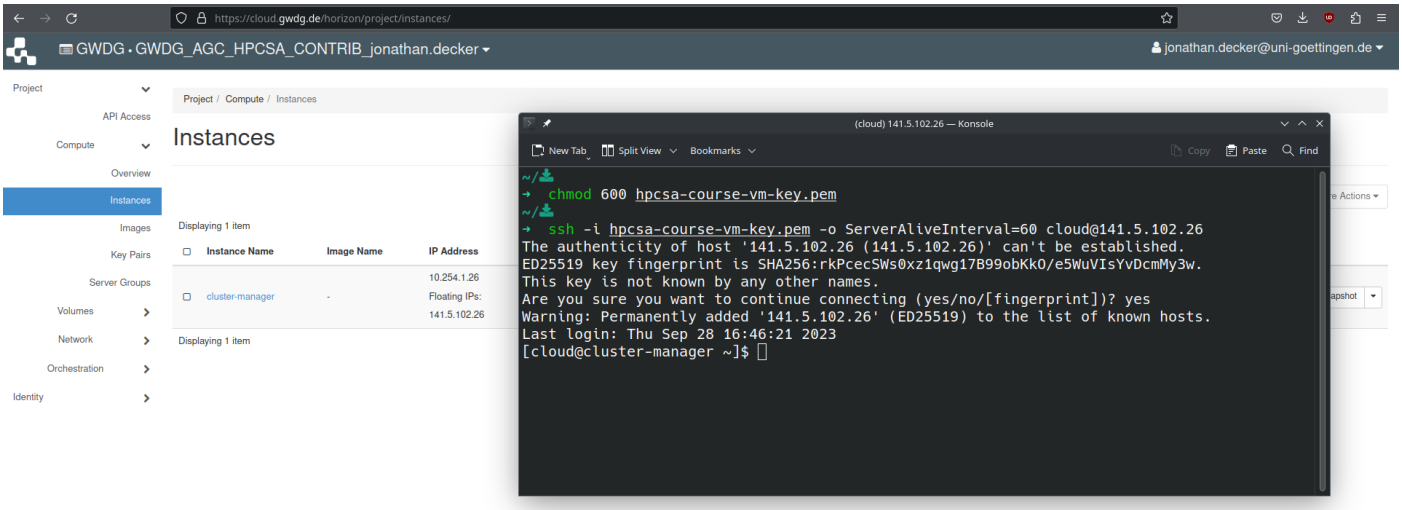

Figure 24: Connection to Cluster-Manager with SSH

MacOS/Linux

1. Search for **Terminal** and open it.
2. Check ssh is provided by running the command `ssh -V`

On MacOS/Linux you need to set the correct permission for the **hpcsa-course-vm-key.pem** key before it can be used with SSH.
Run `chmod 600 hpcsa-course-vm-key.pem`
in the same folder as the key.

On Windows 11 you might get an error about the permission being to open when running SSH with the key. If that happens you can try the following workaround:

Windows 11 SSH Permission Workaround

1. Select **hpcsa-course-vm-key.pem** and open properties. (Shortcut: Alt + Shift)
2. Go to Security → Edit.
3. Remove all users except Administrators.
4. Click on Apply and OK.
5. Now open PowerShell as administrator
6. Run the code below

```
# Set Key File Variable:
New-Variable -Name Key -Value "$env:UserProfile\.ssh\known_hosts"
# Remove Inheritance:
Icacls $Key /c /t /Inheritance:d
# Set Ownership to Owner:
# Key's within $env:UserProfile:
Icacls $Key /c /t /Grant ${env:UserName}:F
```

```
# Key's outside of $env:UserProfile:
TakeOwn /F $Key
Icacls $Key /c /t /Grant:r ${env:UserName}:F
# Remove All Users, except for Owner:
Icacls $Key /c /t /Remove:g Administrator "Authenticated Users"
 BUILTIN\Administrators BUILTIN Everyone System Users
# Verify:
Icacls $Key
# Remove Variable:
Remove-Variable -Name Key
```

Note: If the code above does not work, try again with the full path to the ssh known_hosts file instead of using a variable.

Using SSH

1. In PowerShell or Terminal type the following command
   `ssh -i hpcsa-course-vm-key.pem -o ServerAliveInterval=60 cloud@YOUR_IP`
   where YOUR_IP is the IP address you got earlier.
2. When asked whether you want to continue, type in `yes` .
   See Figure 24 for comparison.
3. Confirm that running `hostname` returns **cluster-manager.novalocal**.

Launch Worker Instances

This section shows how to launch two worker instances at once.

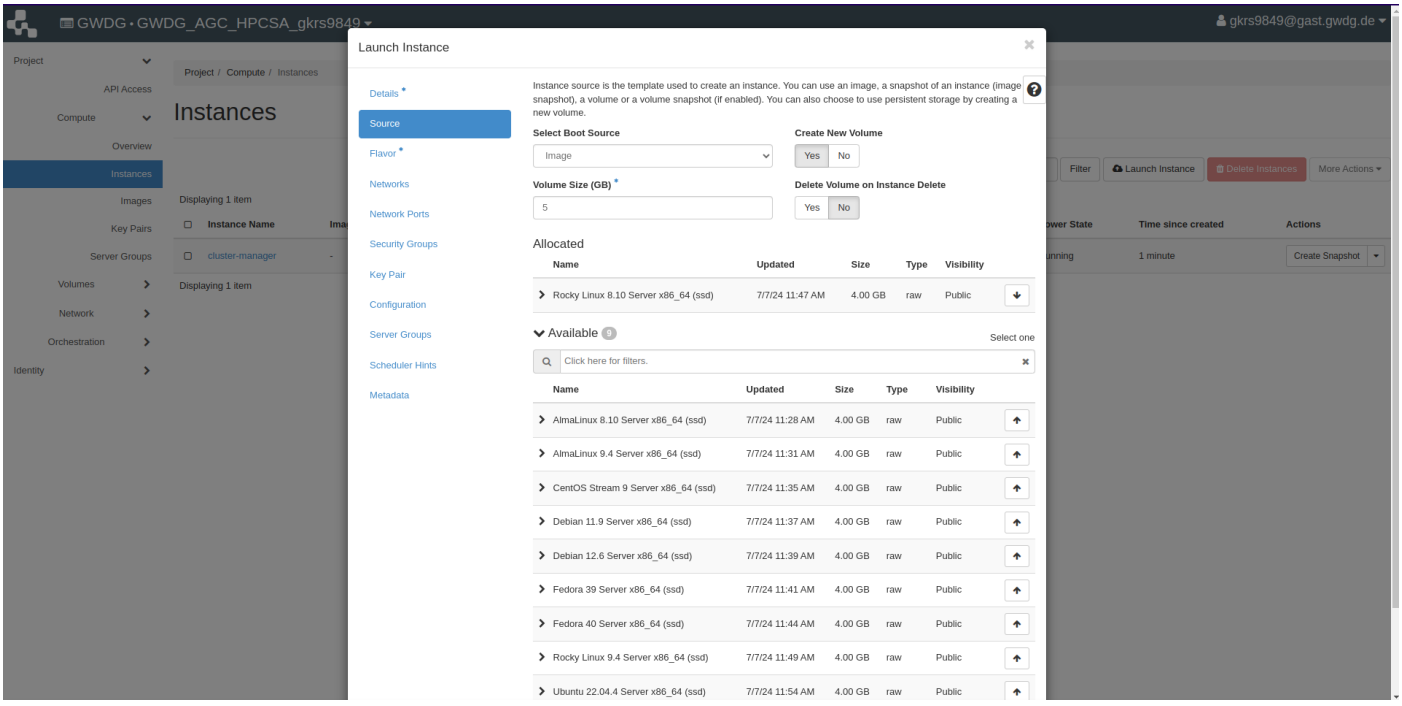

Figure 25: Launch Instance Details for Workers

1. Under the **Compute** tab, select **Instances** and press **Launch Instance**.
2. Set the name to **worker** and the **Count** to **2** as shown in Figure 25.

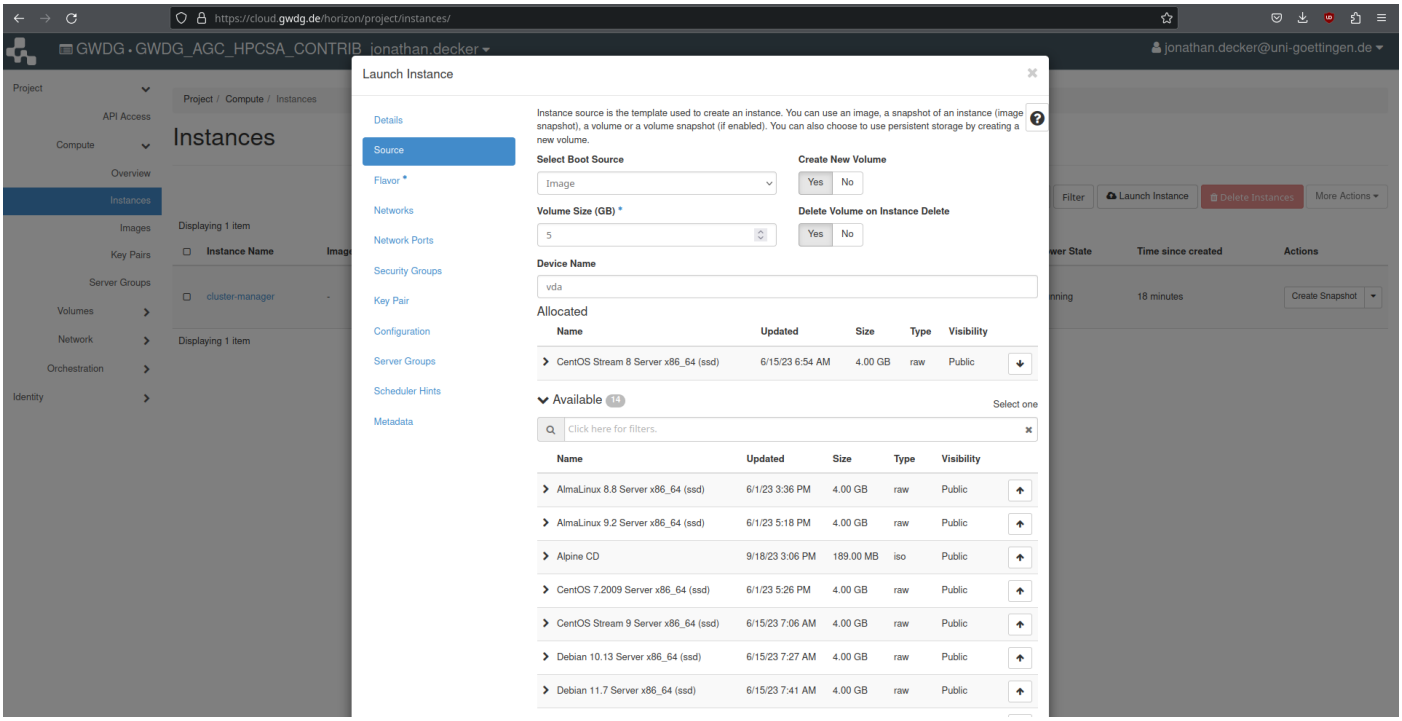

Figure 26: Launch Instance Source for Workers

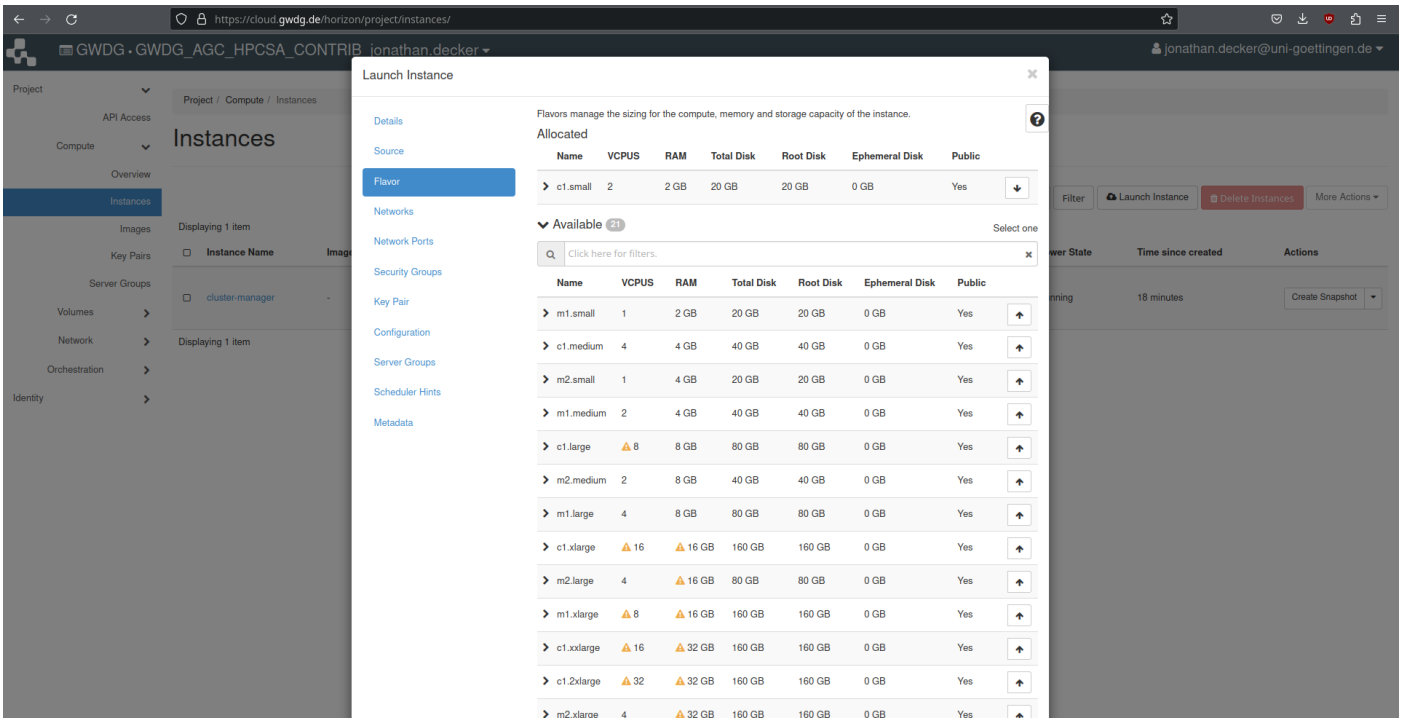

Figure 27: Launch Instance Flavor for Workers

3. Under source set again **Rocky 8** as the OS from the menu at the bottom and both **Create New Volume** and **Delete Volume on Instance Delete** to **Yes** as shown in Figure 26.
4. Move on to **Flavor** and set it to **c1.small** from the list as shown in Figure 27.
5. Proceed with **Security Groups** and add only the **Internal** group as shown in Figure 28.

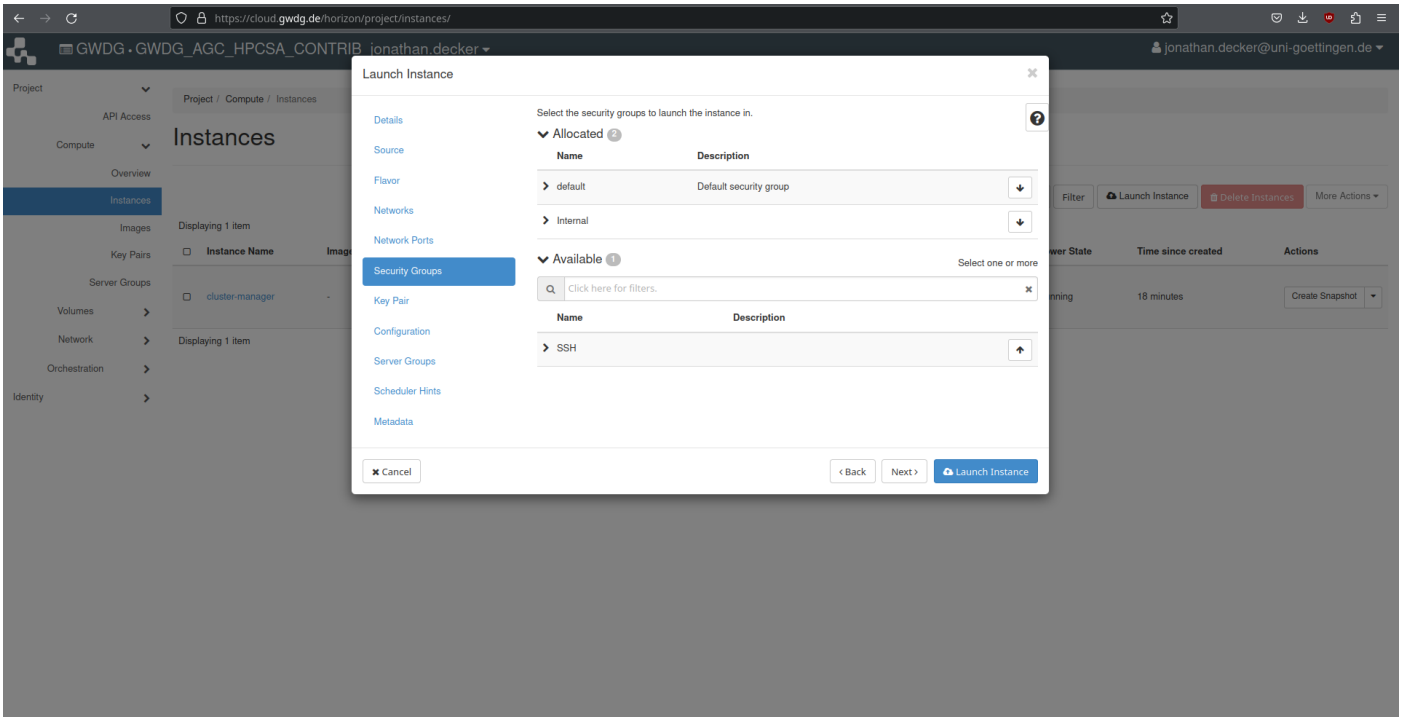

Figure 28: Launch Instance Security Groups for Workers

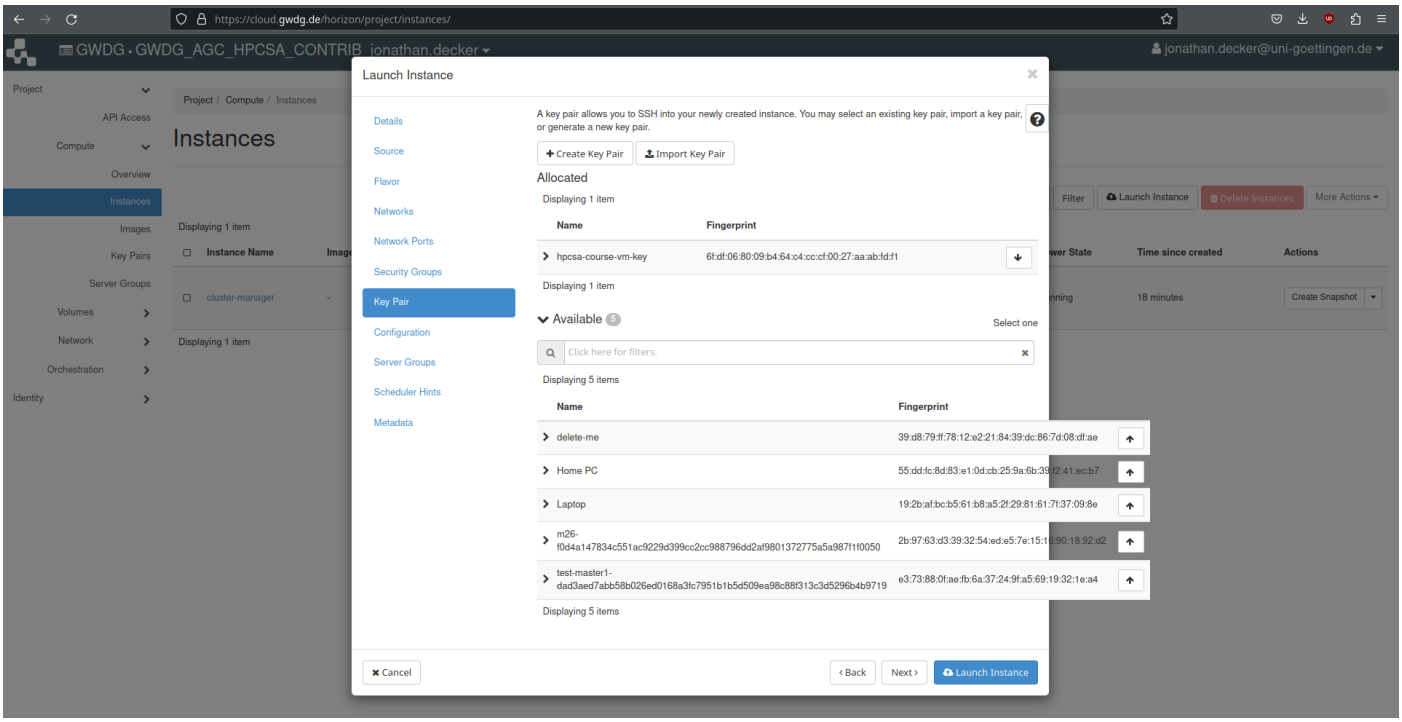

Figure 29: Launch Instance Key Pair for Workers

6. In the next step for **Key Pair** set the **hpcsa-course-vm-key** as shown in Figure 29.
7. Press **Launch Instance** and wait for the system to provision the two worker instances as shown in Figure 30.
8. After a short while the two instances become available and reach the state **Active** as shown in Figure 31.

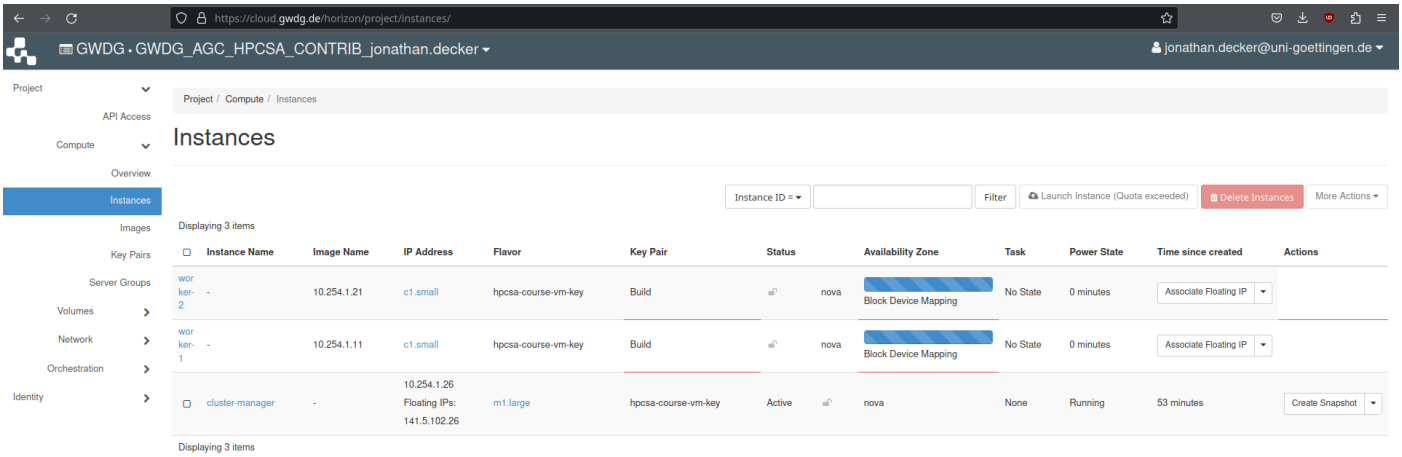

Figure 30: Instance Overview with Workers Launching

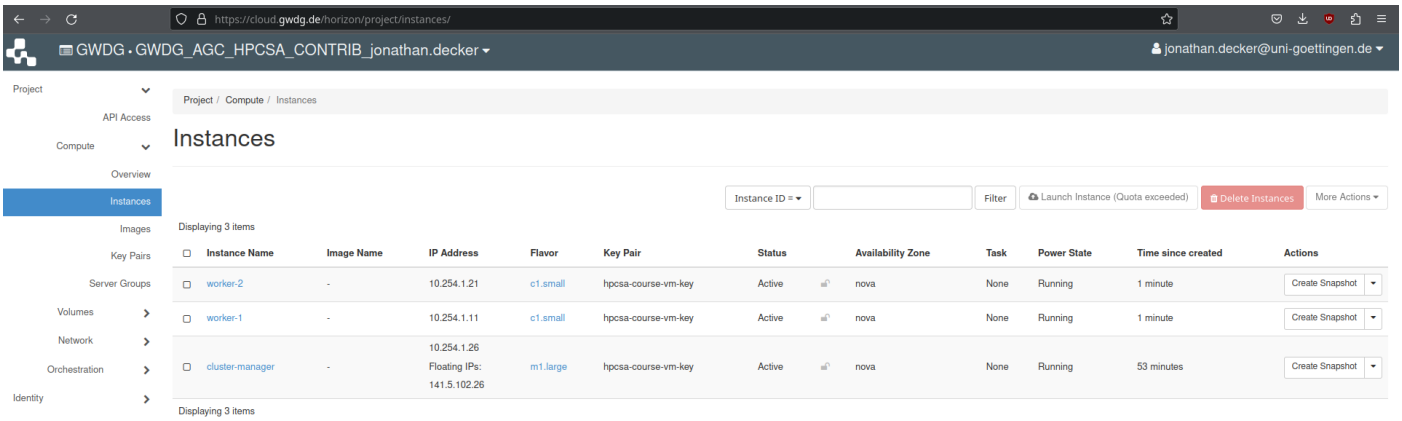

Figure 31: Instance Overview with Workers Running

Connect to Worker Instance via the Main Instance

This section shows how to connect to your **worker** instances by jumping through your **cluster-manager**.

1. Note down the IP addresses of the worker nodes from the instances overview as visible in Figure 31.
2. In PowerShell or Terminal type the following command:

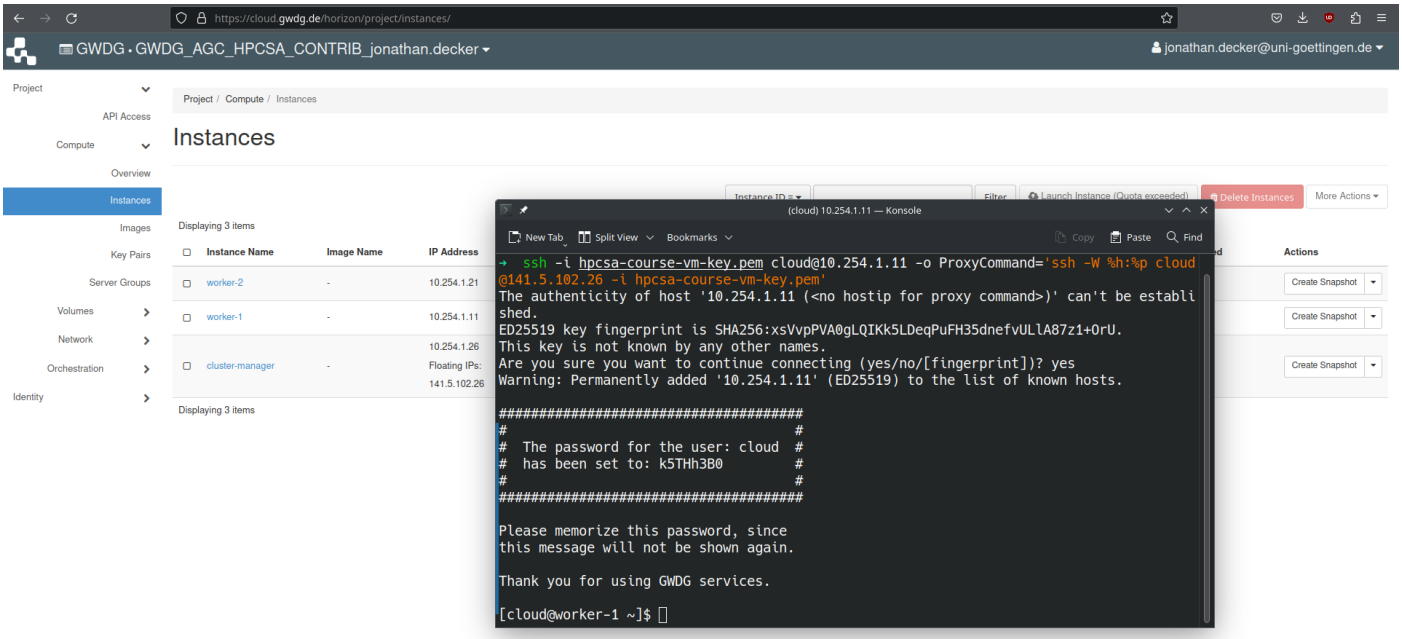

Figure 32: Connected to Worker through Cluster-Manager via SSH

```
ssh -i hpcsa-course-vm-key.pem cloud@YOUR_WORKER_IP
 -o ProxyCommand='ssh -W %h:%p cloud@YOUR_FLOATING_IP
 -i hpcsa-course-vm-key.pem'
```

where YOUR_WORKER_IP is the IP address of one of your **workers** and YOUR_FLOATING_IP is the floating IP address of the **cluster-manager**. See for comparison Figure 32.
By switching the IP of the workers in the command, you can connect to the other worker.

3 Useful Commands

Rebooting Instances

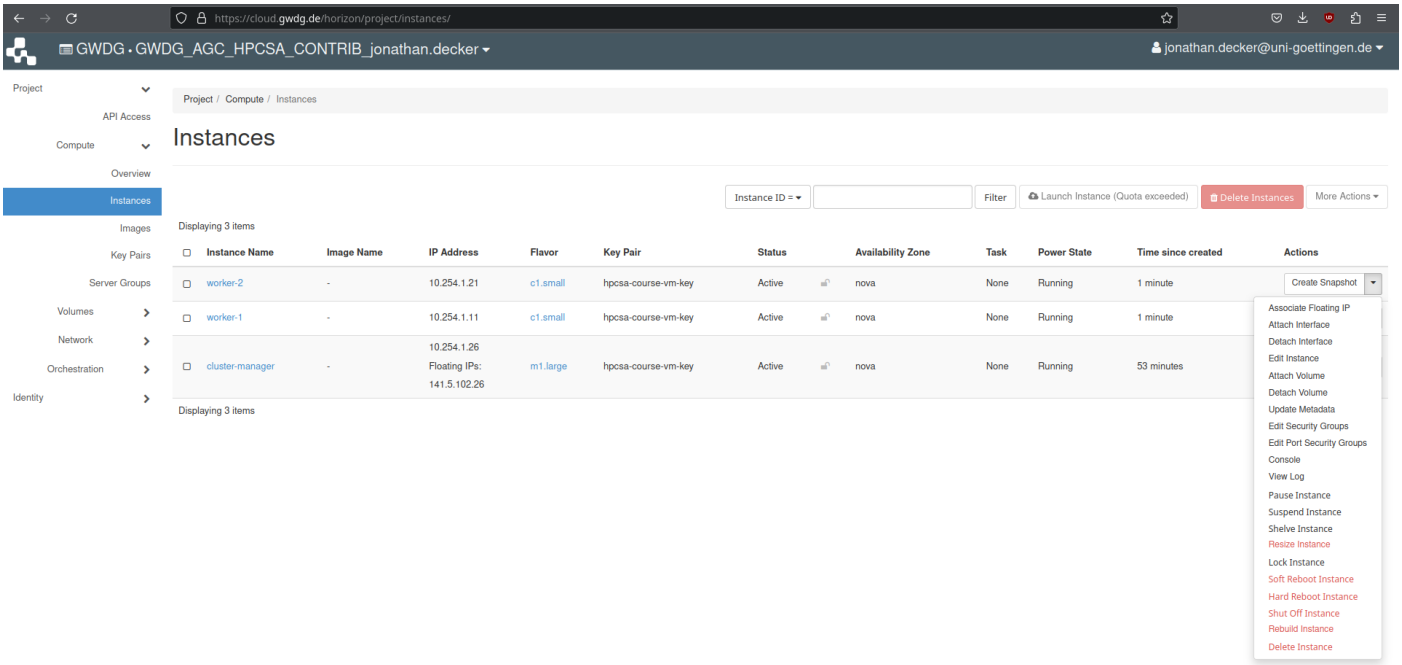

Figure 33: Instance Overview Actions Drop-Down Menu Reboot

When working with VMs, it might be necessary to reboot them via an external command.
Under **Compute** tab, select **Instances** and find the **Actions** drop-down menu for each of your instances. This menu contains the options **Soft Reboot Instance** and **Hard Reboot Instance**, which cause an instance to restart. See for reference Figure 33.

SSH Port Forwarding

When deploying applications on one of your VMs that expose a web interface, e.g., a web page, you might need to open said interface in a web browser on your workstation. For this you have two options, you either open the port as shown in the instructions above for the SSH port, or you use SSH port forwarding. Opening the port, especially, globally, poses certain security risks and should not be done casually. This option also only works when the application is running on a port on your **cluster-manager** and not on one of your **workers**.

The alternative is to use SSH port forwarding. You can forward a port from your cluster-manager to your workstation as follows:

```
ssh -L REMOTE_PORT:FLOATING_IP:LOCAL_PORT -o ServerAliveInterval=60
 -i hpcsa-course-vm-key.pem cloud@FLOATING_IP
```

Use the floating IP of the **cluster-manager** as FLOATING_IP and the port the application is using on the **cluster-manger** as REMOTE_PORT. As LOCAL_PORT you can use any port that is open on your workstation, even the same number as remote port. Please note that on Linux machines, the port numbers up to 1024 are typically handled as privileged ports so assigned any of them will require root privileges. Alternatively, you can use a higher port locally such as having the remote port 80 forwarded to your local port 8080.
While the port forward is running, you should be able to access the forwarded application under `localhost:LOCAL_PORT`.

It is also possible to forward the port of a worker to your workstation as follows:

```
ssh -L REMOTE_PORT:YOUR_WORKER_IP:LOCAL_PORT -i hpcsa-course-vm-key.pem cloud@YOUR_WORKER_IP
 -o ProxyCommand='ssh -W %h:%p cloud@YOUR_FLOATING_IP
 -i hpcsa-course-vm-key.pem'
```

Another version of this command that port forwards the remote port to your workstation in the background goes as follows:

```
ssh -N -L REMOTE_PORT:FLOATING_IP:LOCAL_PORT -o ServerAliveInterval=60
 -i hpcsa-course-vm-key.pem cloud@FLOATING_IP >/dev/null 2>&1 &
```

The `-N` flag tells the system to not open a shell, the `>/dev/null 2>&1` ensures that messages from the port forwarding are not send to the shell and the final `&` sends SSH into the background. Alternatively the `-f` flag does the same as the final `&` .

File Transfer

You can use the `scp` command to transfer files from your system to one of your VMs and from one of your VMs to your local system as follows:
To upload to **cluster-manager**:

```
scp -i hpcsa-course-vm-key.pem LOCAL_FILE cloud@FLOATING_IP:/home/cloud
```

Instead of `/home/cloud` you can specify another location for your file on the VM.
To download from **cluster-manager**:

```
scp -i hpcsa-course-vm-key.pem cloud@FLOATING_IP:PATH_TO_THE_FILE .
```

The `.` specifies the location where to place the downloaded file and means in the current folder. To download from a **worker**:

```
scp -i hpcsa-course-vm-key.pem LOCAL_FILE cloud@WORKER_IP:/home/cloud
 -o ProxyCommand='ssh -W %h:%p cloud@YOUR_FLOATING_IP
 -i hpcsa-course-vm-key.pem'
```

To upload to a **worker**:

```
scp -i hpcsa-course-vm-key.pem cloud@WORKER_IP:PATH_TO_THE_FILE .
 -o ProxyCommand='ssh -W %h:%p cloud@YOUR_FLOATING_IP
 -i hpcsa-course-vm-key.pem'
```