Cloud Computing – WS 2017

Exercise 1 : Cloud Access and Security

7th November 2017

Anshul Jindal

anshul.jindal@tum.de

Index

Introduction

Cloud Service Providers

What?

- are companies that offers network services, infrastructure, or business applications in the cloud.
- The cloud services are hosted in a data centre than can be accessed by companies or individuals using network connectivity.

Who ?

Cloud Service Provider	IAAS	PAAS	SAAS
Amazon (AWS)	X	X	
Century Link	X	X	
Google (GCP)	X	X	X
IBM (Bluemix)	X	X	X
Microsoft (Azure)	X	X	X
Rackspace	X	X	
Salesforce.com		X	X
SAP	X	X	X

The LRZ Compute Cloud

- Cloud service offering from LRZ(Leibniz-Rechenzentrum)
- Infrastructure as a Service (IaaS)
- **Based upon OpenNebula**
  - **Wiki Definition of OpenNebula: It** is a cloud computing platform for managing heterogeneous distributed data centre infrastructures. The OpenNebula platform manages a data centre's virtual infrastructure to build private, public and hybrid implementations of infrastructure as a service.
- Login: https://www.cloud.mwn.de/

All the exercises will be done on it.

Cloud Security

- **Cloud computing security** or **cloud security** refers to a set of policies, technologies, and controls deployed to protect data, applications, and the associated infrastructure of cloud computing.
- The biggest cloud security issue is **Authentication.**
  - **VM/Cloud Level Authentication**
    - User Level Authentication
    - Group Level Authentication
  - **Service Level Authentication**
    - Global Authentication
    - Single Route Authentication

Exercise involves usage of all the above authentications.

Node.js

What?

- It is an application runtime environment that allows you to write server-side applications in JavaScript.
- Its unique I/O model allows creation of scalable and real-time solutions.

When to use ?

- **Real time applications :** applications that have to process a high volume of short messages requiring low latency
- **Fast and scalable environment:**
  - ability to process many requests with low response times.
  - makes it a great fit for modern web applications that carry out lots of processing on the client's side. Ex. single-page applications

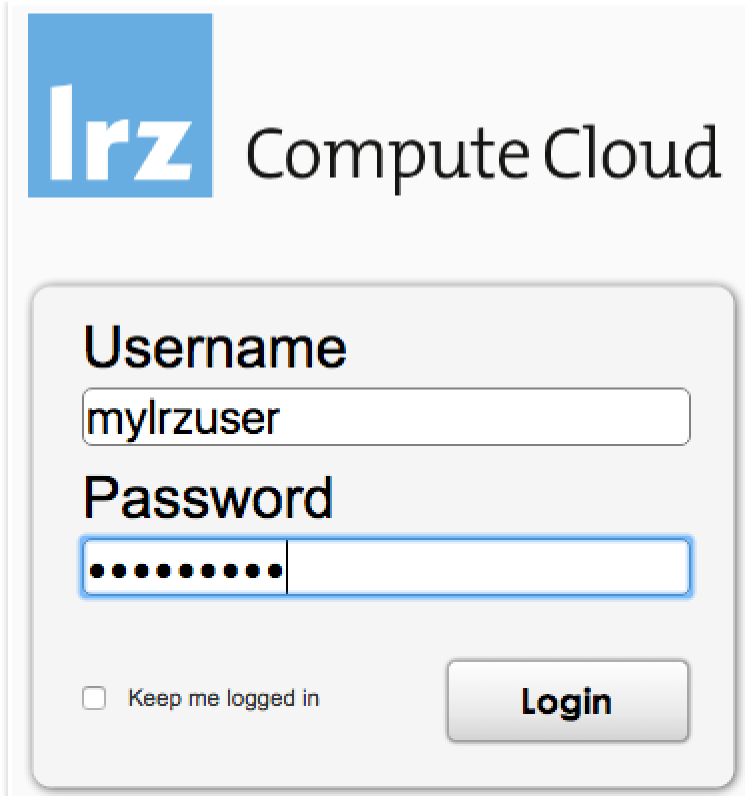
When not to use ?

- **CPU-heavy jobs:**
  - Node.js is based on an event-driven, non-blocking I/O model, and uses only a single CPU core.
  - CPU-heavy operations will just block incoming requests, rendering the biggest advantage of Node.js useless
- **CRUD: U**sing Node.js would be superfluous for simple HTML or CRUD applications in which you don't need a separate API, and all data comes directly from the server.

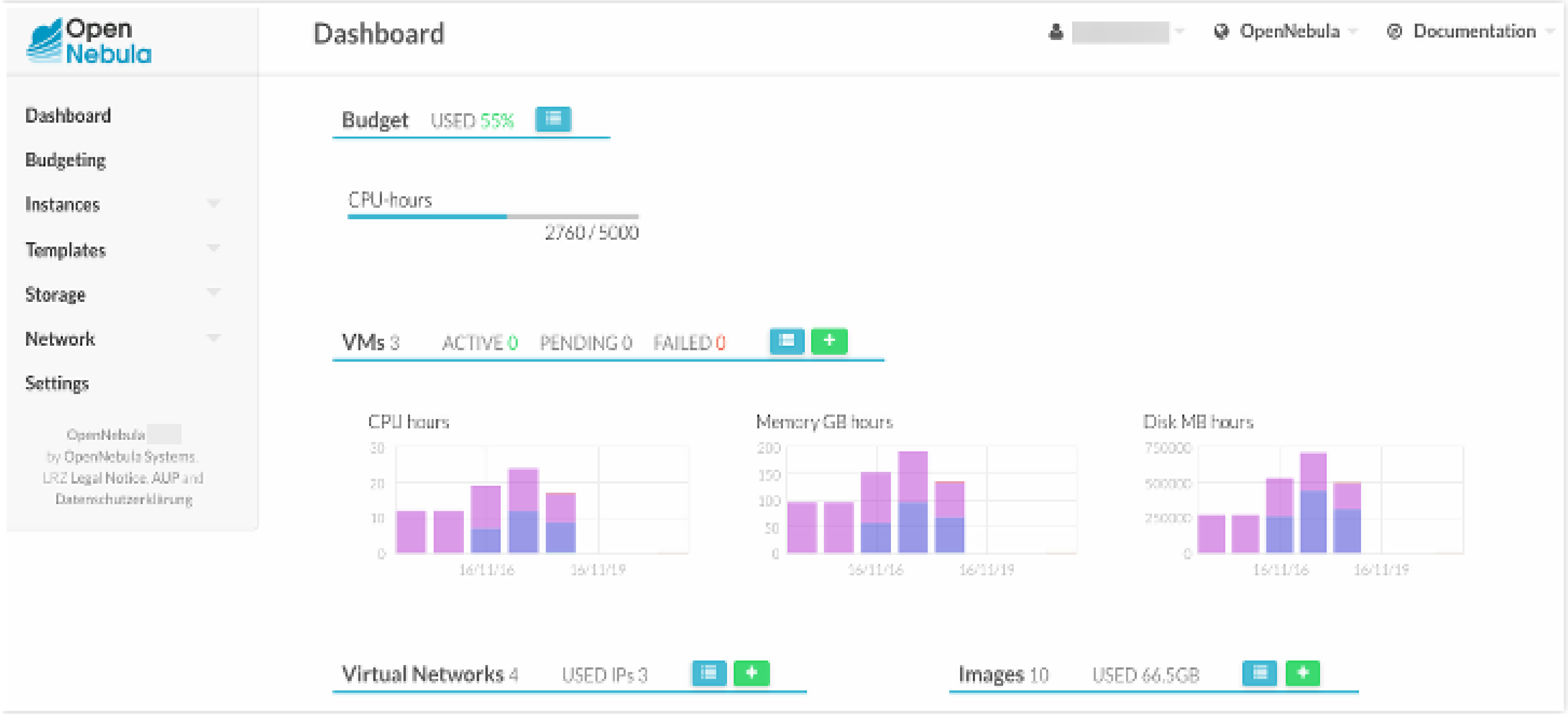
LRZ Account Access and VM Creation

Step1: Cloud Account Login

Login with your given credentials to : https://www.cloud.mwn.de



Step2: Cloud Account Dashboard



Budget USED 55%

CPU-hours

2760 / 5000

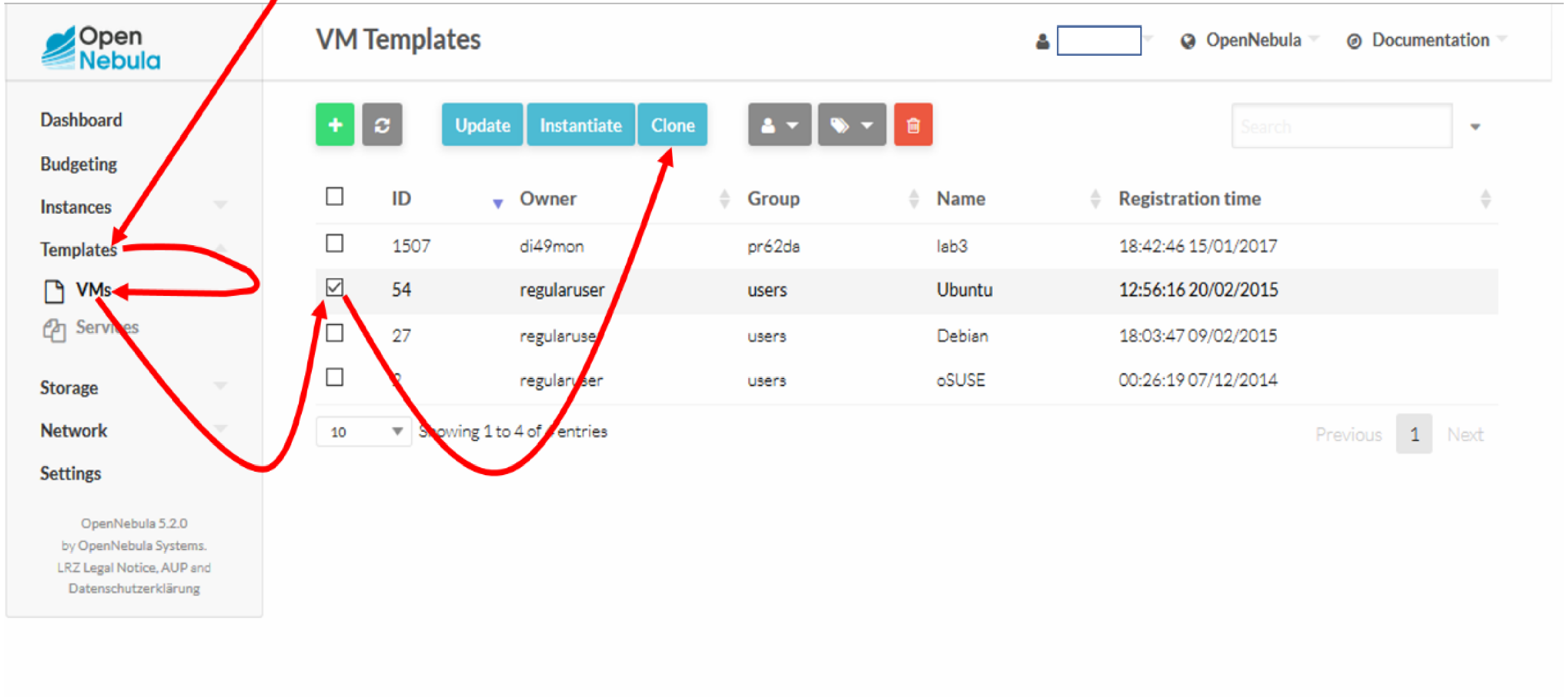
VMs 3 ACTIVE 0 PENDING 0 FAILED 0

Virtual Networks 4 USED IPs 3

Images 10 USED 66.5GB

OpenNebula [redacted]
by OpenNebula Systems.
LRZ Legal Notice, AUP and
Datenschutzerklärung

Step3: Creation of VM template

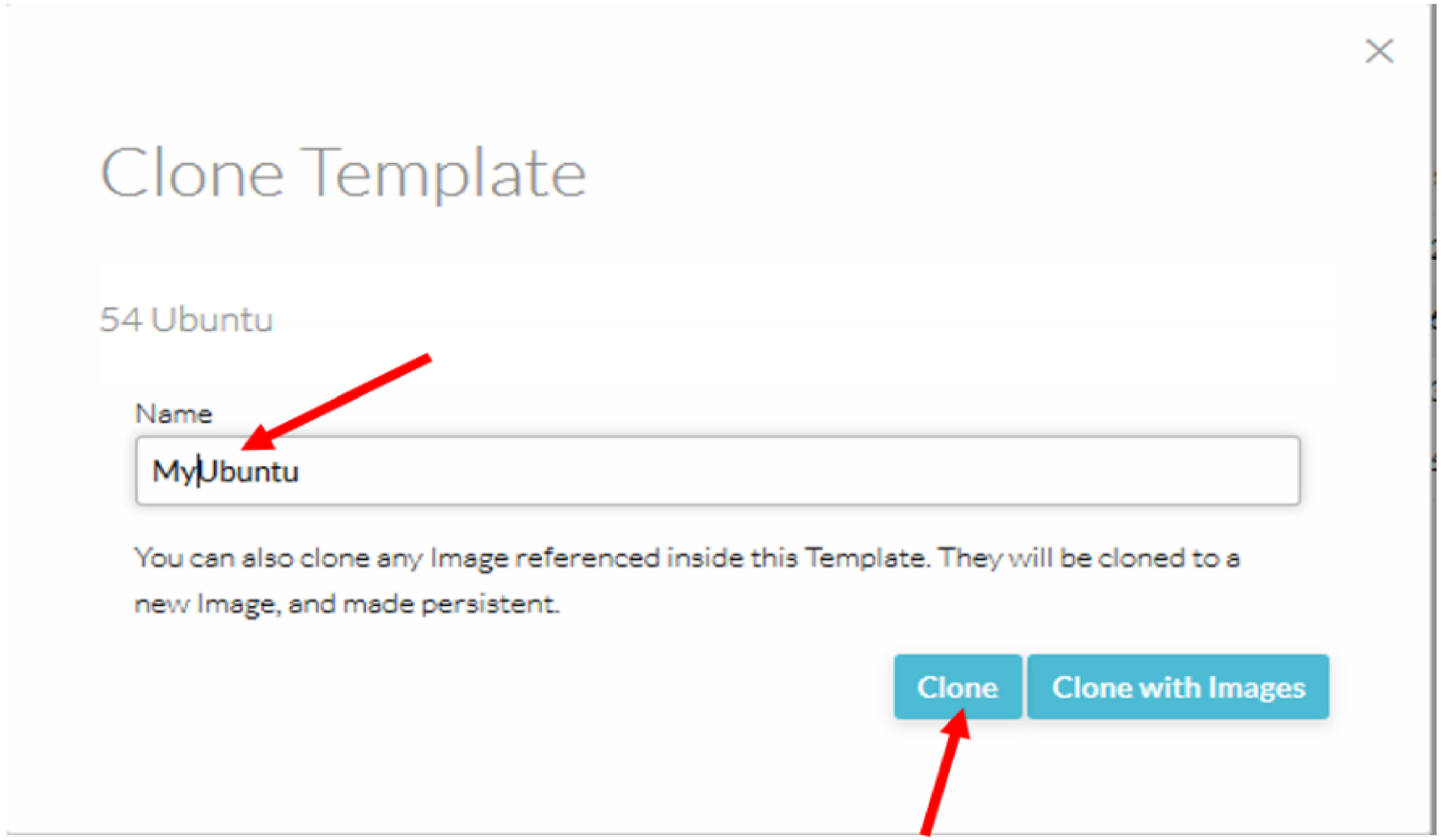


	ID	Owner	Group	Name	Registration time
☐	1507	di49mon	pr62da	lab3	18:42:46 15/01/2017
☑	54	regularuser	users	Ubuntu	12:56:16 20/02/2015
☐	27	regularuser	users	Debian	18:03:47 09/02/2015
☐	0	regularuser	users	oSUSE	00:26:19 07/12/2014

Showing 1 to 4 of 4 entries

OpenNebula 5.2.0
by OpenNebula Systems.
LRZ Legal Notice, AUP and
Datenschutzerklärung

Step3: Creation of VM template Cont..



Clone Template

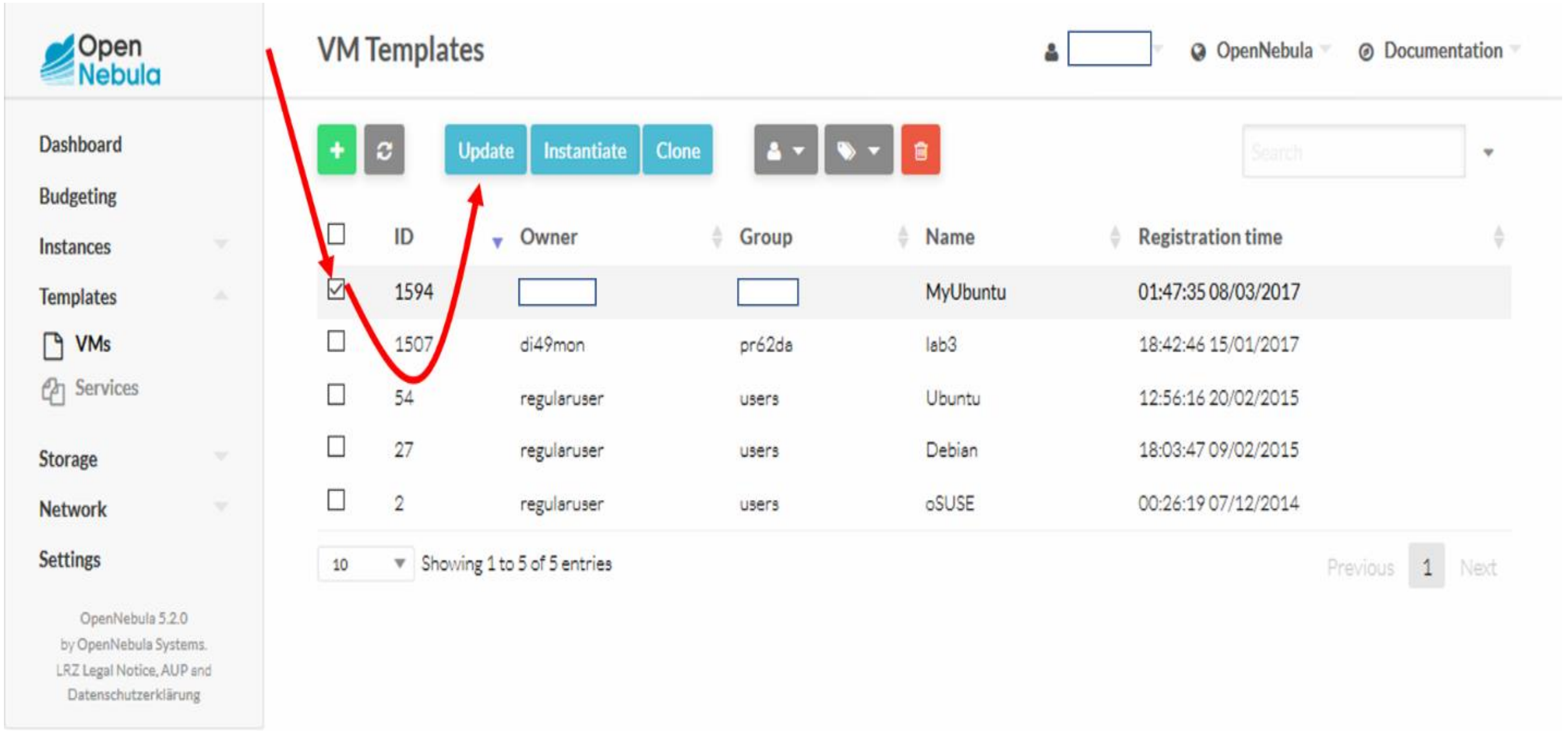
54 Ubuntu

Name

MyUbuntu

You can also clone any Image referenced inside this Template. They will be cloned to a new Image, and made persistent.

Step4: Updating cloned VM template

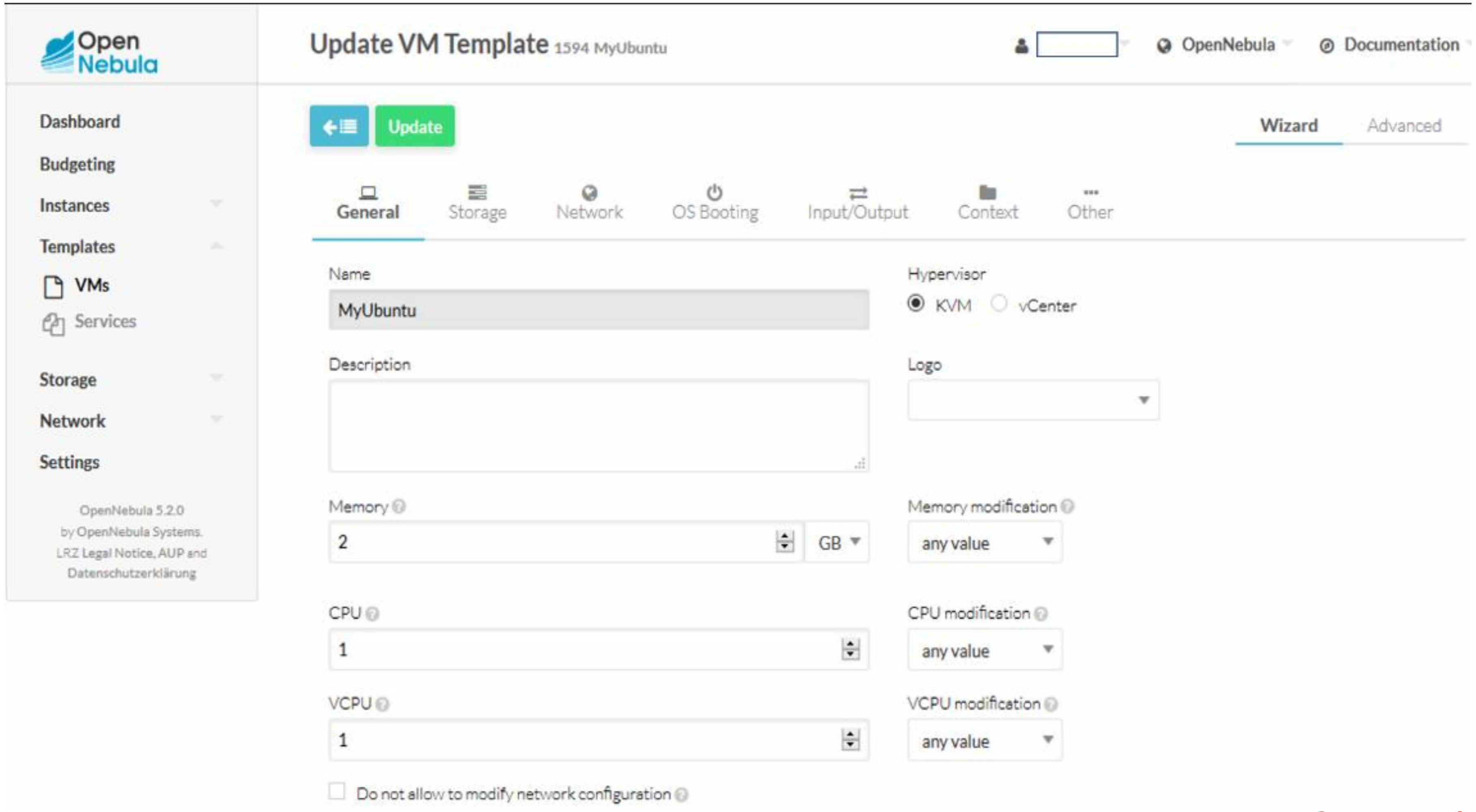


	ID	Owner	Group	Name	Registration time
☑	1594	[redacted]	[redacted]	MyUbuntu	01:47:35 08/03/2017
☐	1507	di49mon	pr62da	lab3	18:42:46 15/01/2017
☐	54	regularuser	users	Ubuntu	12:56:16 20/02/2015
☐	27	regularuser	users	Debian	18:03:47 09/02/2015
☐	2	regularuser	users	oSUSE	00:26:19 07/12/2014

10 Showing 1 to 5 of 5 entries

OpenNebula 5.2.0
by OpenNebula Systems.
LRZ Legal Notice, AUP and
Datenschutzerklärung

Step4: Updating cloned VM template Cont..



Update VM Template 1594 MyUbuntu

Name: MyUbuntu

Hypervisor: [✓ KVM] vCenter

Description:

Logo:

Memory: 2 GB

Memory modification: any value

CPU: 1

CPU modification: any value

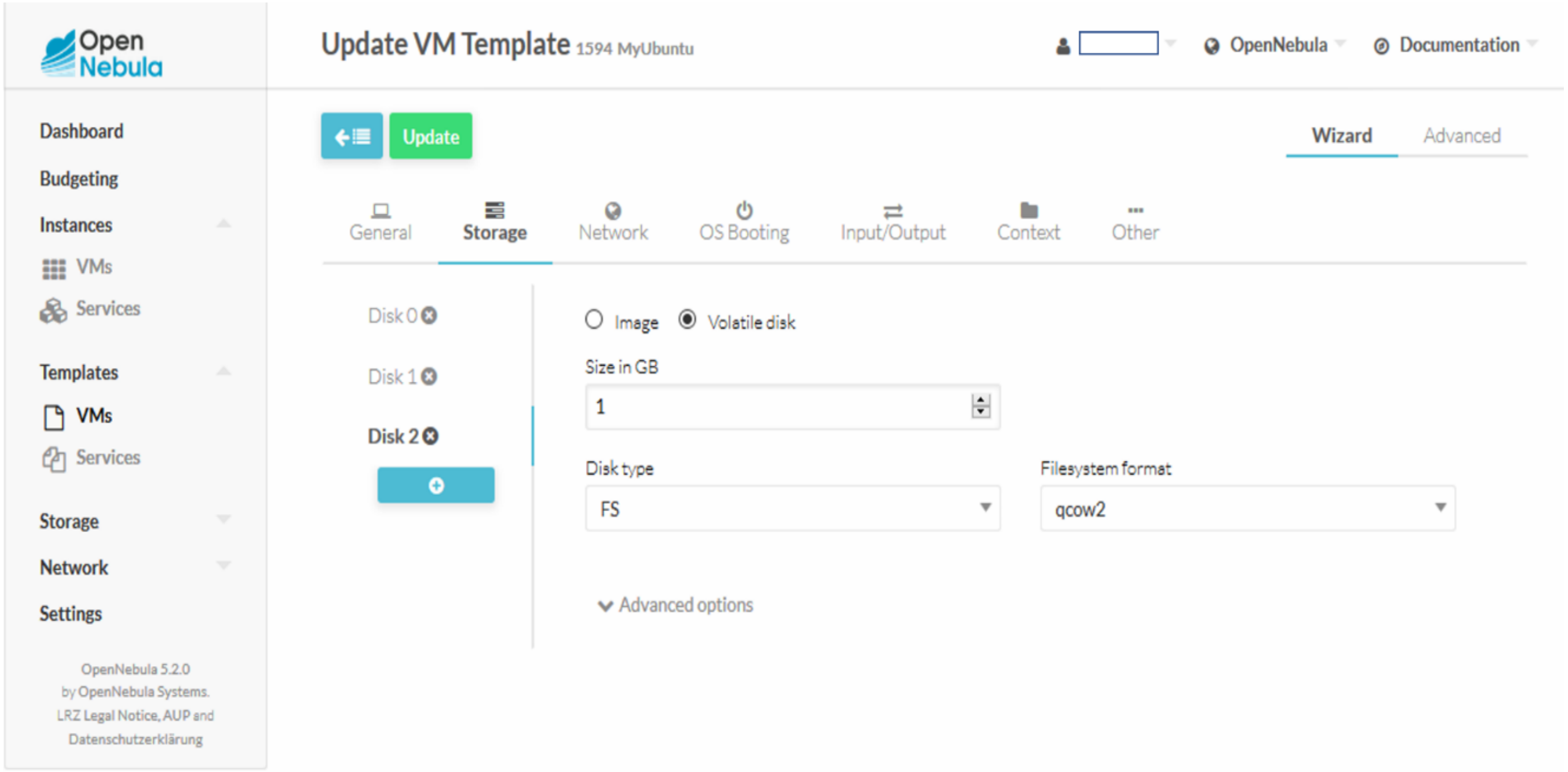
VCPU: 1

VCPU modification: any value

☐ Do not allow to modify network configuration

Left Everything to default

Step4: Updating cloned VM template Cont..



Update VM Template 1594 MyUbuntu

( ) Image [✓ Volatile disk]

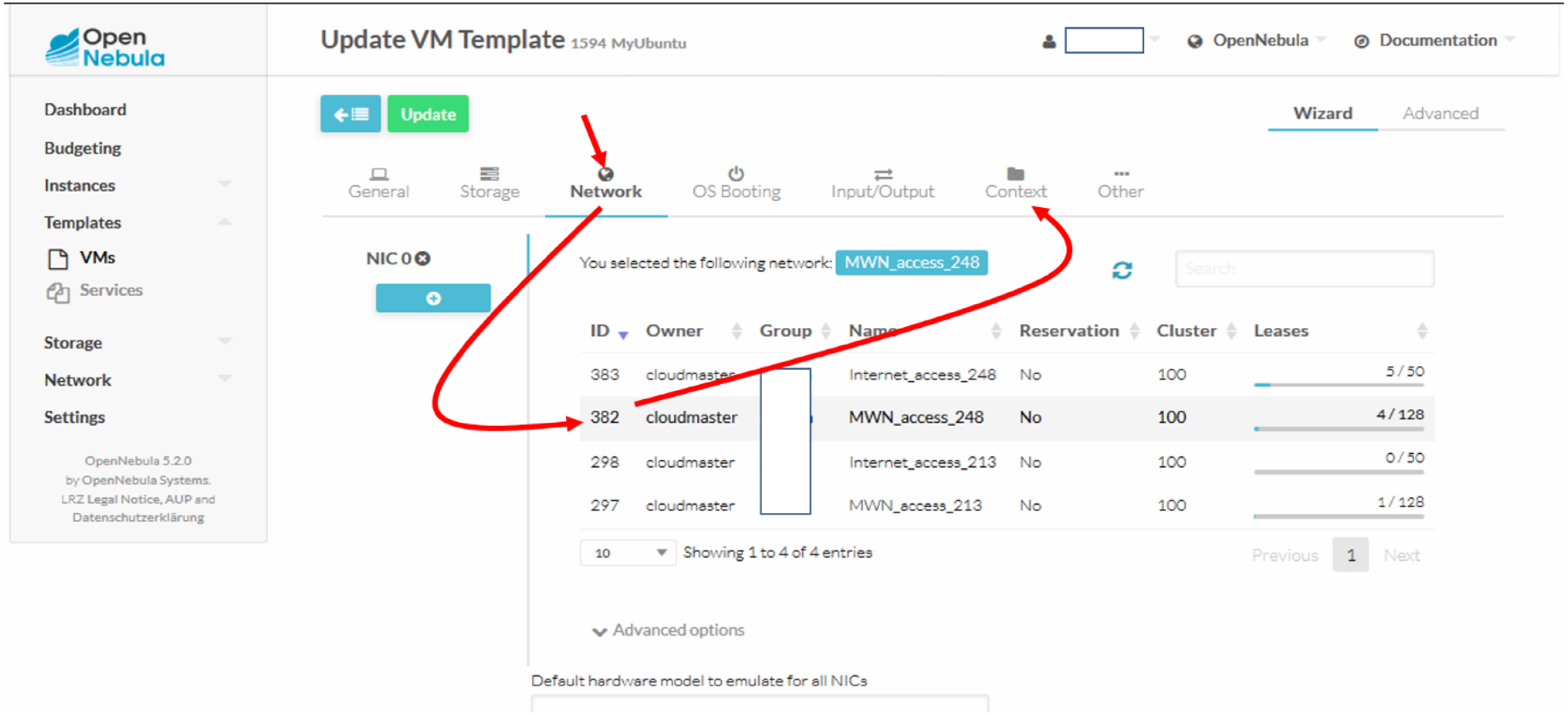
Size in GB: 1

Disk type: FS

Filesystem format: qcow2

Left Everything to default

Step4: Updating cloned VM template Cont..



Update VM Template 1594 MyUbuntu

OpenNebula 5.2.0 by OpenNebula Systems. LRZ Legal Notice, AUP and Datenschutzerklärung

You selected the following network: MWN_access_248

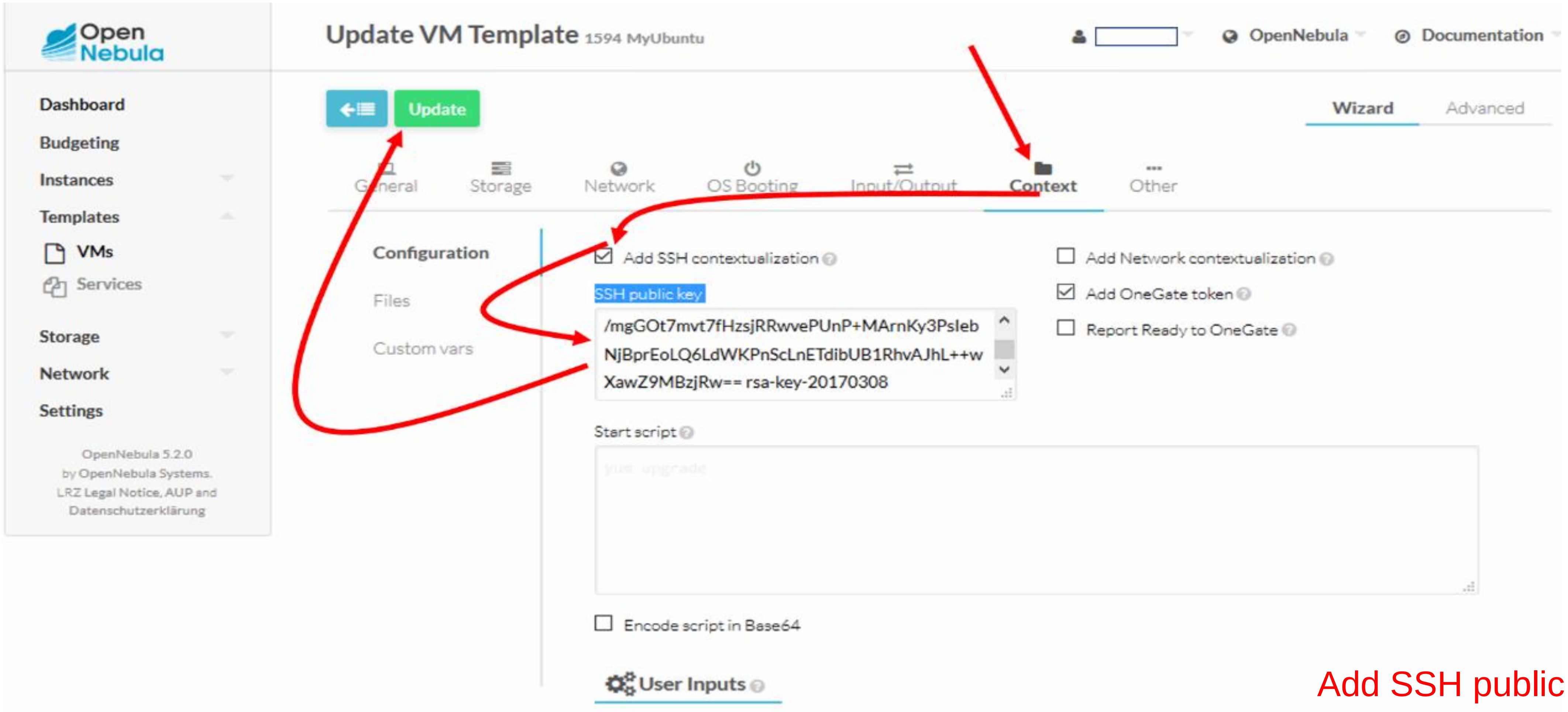
ID	Owner	Group	Name	Reservation	Cluster	Leases
383	cloudmaster		Internet_access_248	No	100	5 / 50
382	cloudmaster		MWN_access_248	No	100	4 / 128
298	cloudmaster		Internet_access_213	No	100	0 / 50
297	cloudmaster		MWN_access_213	No	100	1 / 128

10 Showing 1 to 4 of 4 entries Previous 1 Next

Default hardware model to emulate for all NICs

Add Network interface

Step4: Updating cloned VM template Cont..



[✓] Add SSH contextualization

SSH public key

```
/mgGOt7mvt7fHzsjRRwvePUnP+MArnKy3Psleb
NjBprEoLQ6LdWKPnScLnETdibUB1RhvAJhL++w
XawZ9MBzjRw== rsa-key-20170308
```

[ ] Add Network contextualization

[✓] Add OneGate token

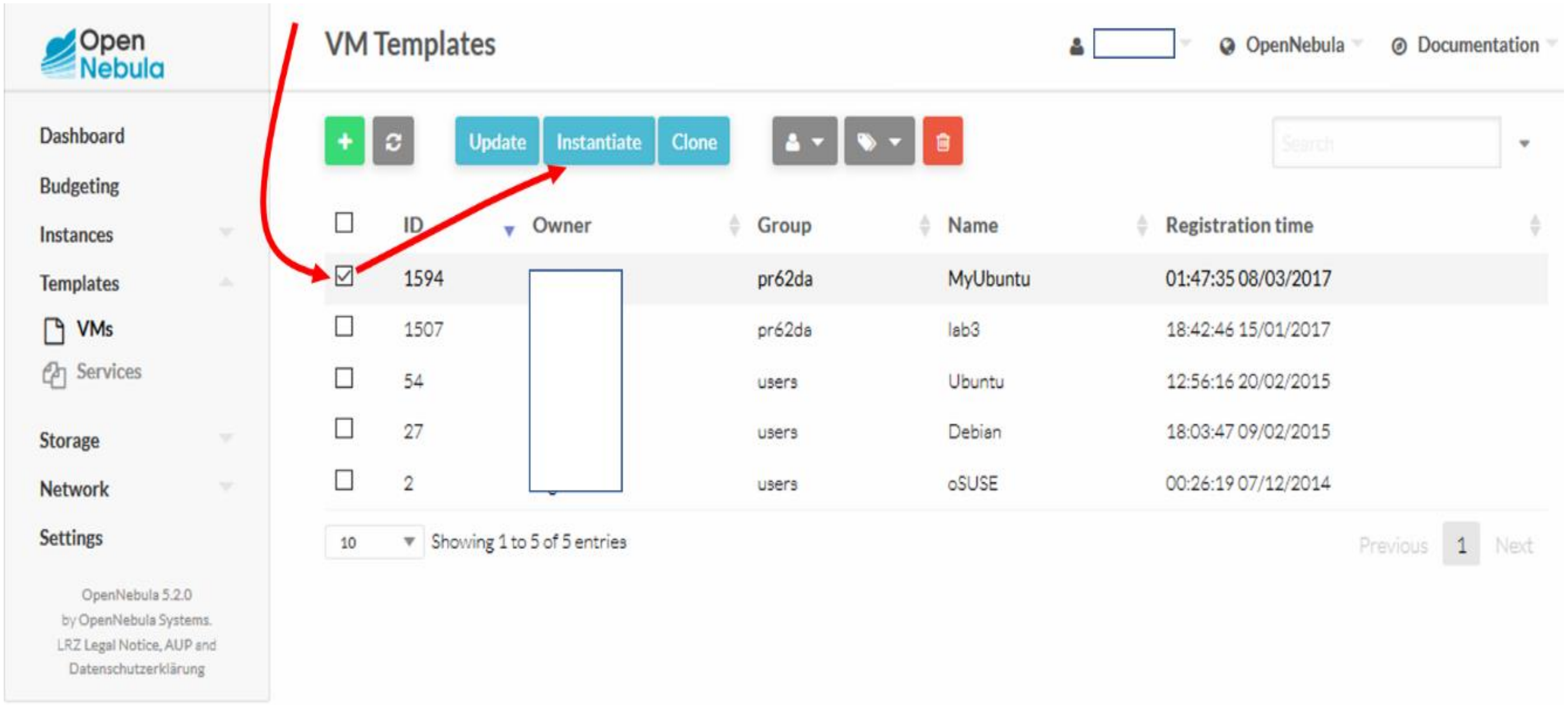
[ ] Report Ready to OneGate

Start script

[ ] Encode script in Base64

Add SSH public key for login
(You need to first create one)
ssh-keygen -t rsa

Step5: Instantiate VM template



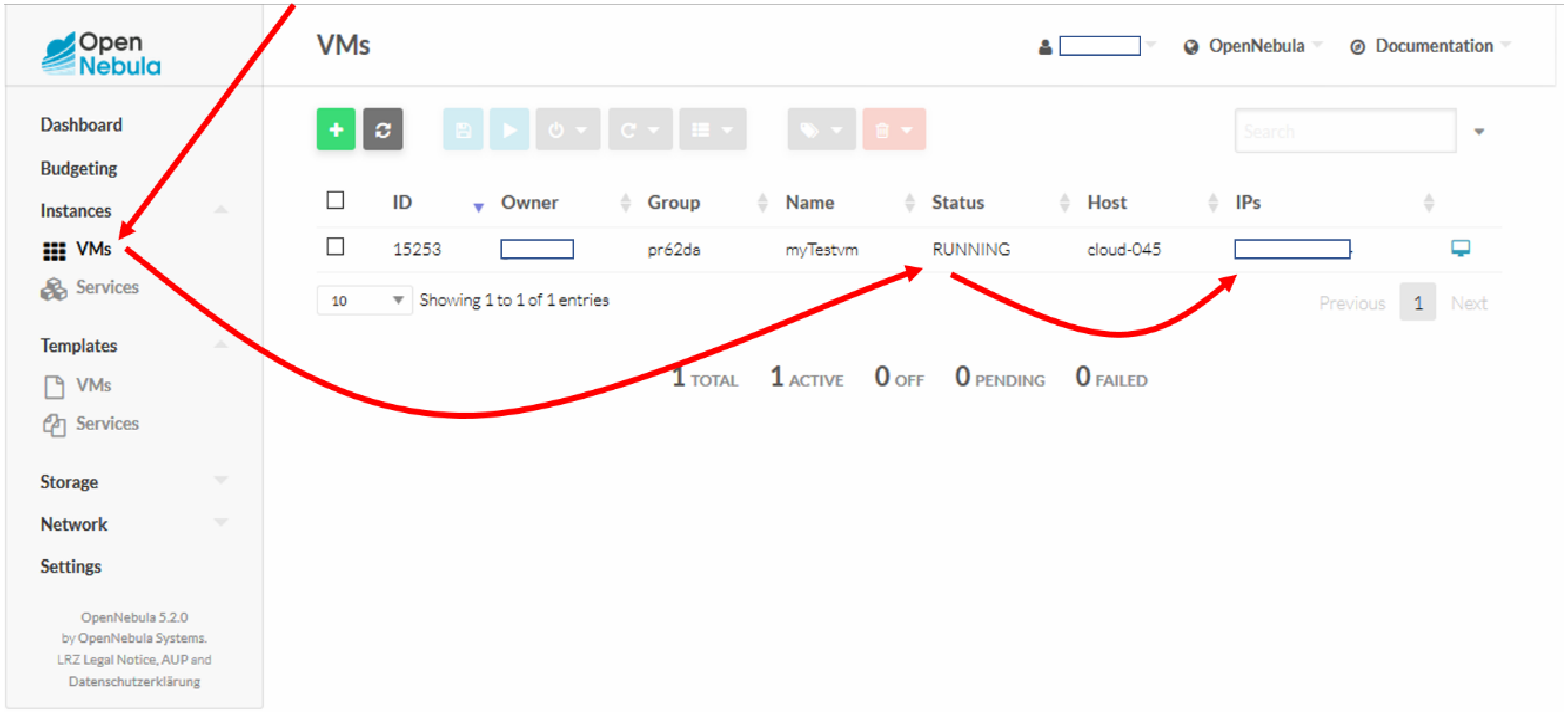
	ID	Owner	Group	Name	Registration time
☑	1594	[redacted]	pr62da	MyUbuntu	01:47:35 08/03/2017
☐	1507	[redacted]	pr62da	lab3	18:42:46 15/01/2017
☐	54	[redacted]	users	Ubuntu	12:56:16 20/02/2015
☐	27	[redacted]	users	Debian	18:03:47 09/02/2015
☐	2	[redacted]	users	oSUSE	00:26:19 07/12/2014

10 Showing 1 to 5 of 5 entries

Previous 1 Next

OpenNebula 5.2.0
by OpenNebula Systems.
LRZ Legal Notice, AUP and
Datenschutzerklärung

Step5: Instantiate VM template Cont..



	ID	Owner	Group	Name	Status	Host	IPs
☐	15253	[redacted]	pr62da	myTestvm	RUNNING	cloud-045	[redacted]

10 Showing 1 to 1 of 1 entries

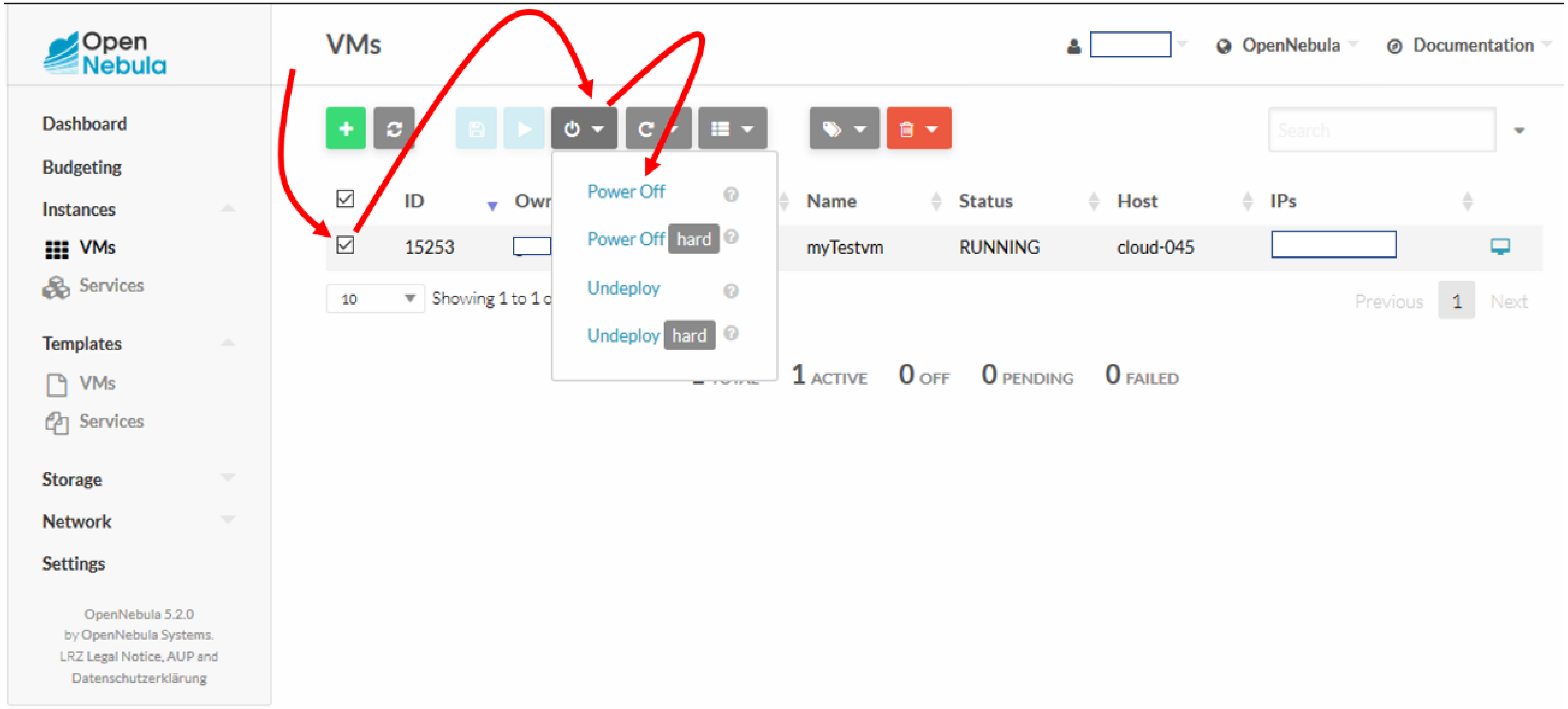
1 TOTAL 1 ACTIVE 0 OFF 0 PENDING 0 FAILED

OpenNebula 5.2.0
by OpenNebula Systems.
LRZ Legal Notice, AUP and
Datenschutzerklärung

Step5: SSH into your VM

```
ssh -i Key_Path root@IP_Address
```

Step6: Power Off/Undeploy



ID	Name	Status	Host
15253	myTestvm	RUNNING	cloud-045

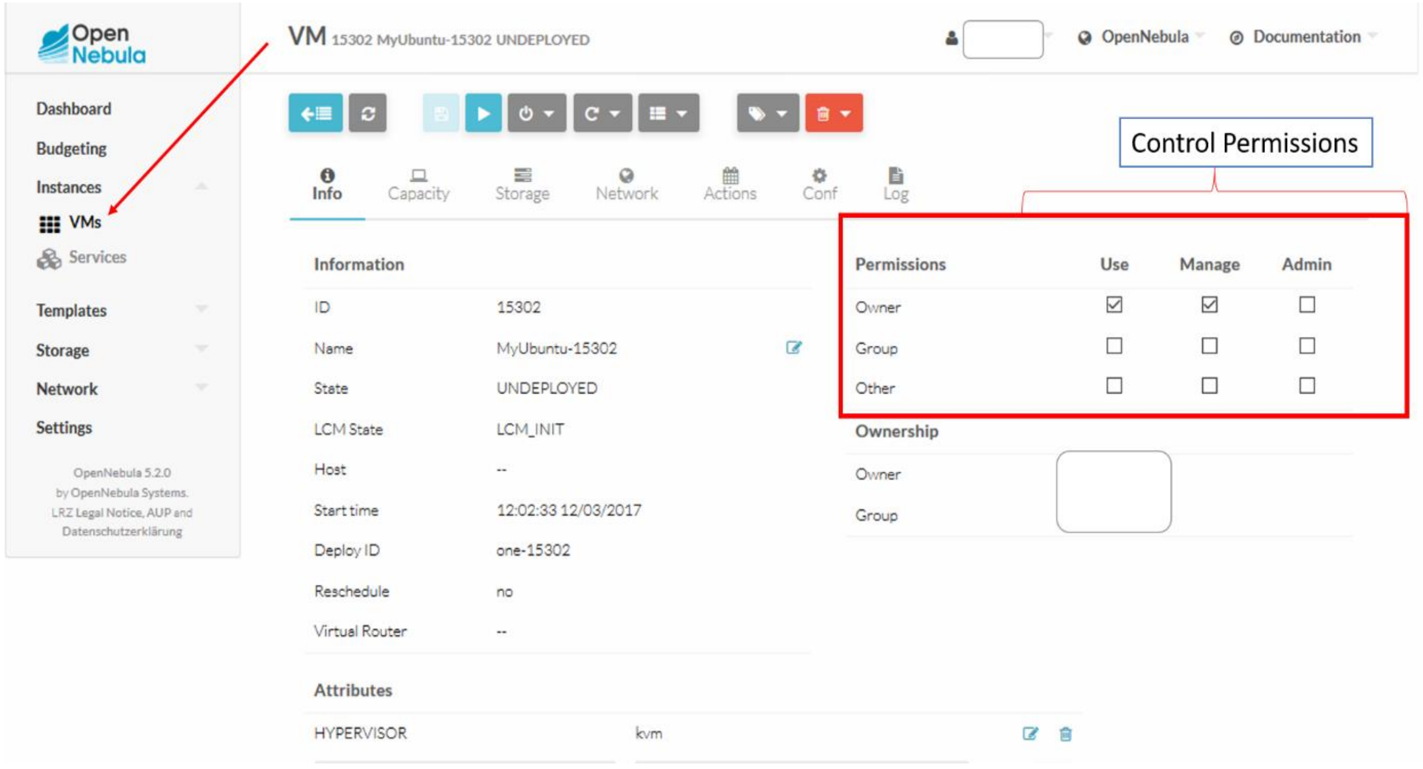
Please Undeploy if not needed anymore

Cloud Security

VM/Cloud Level Authentication

- **User Level Authentication:**
  - This task involves to add the users and give them access permissions to the VM.
  - Can be done using some Linux commands.

- **Group Level Authentication**



OpenNebula 5.2.0 by OpenNebula Systems. LRZ Legal Notice, AUP and Datenschutzerklärung

VM 15302 MyUbuntu-15302 UNDEPLOYED

Information	
ID	15302
Name	MyUbuntu-15302
State	UNDEPLOYED
LCM State	LCM_INIT
Host	--
Start time	12:02:33 12/03/2017
Deploy ID	one-15302
Reschedule	no
Virtual Router	--

Permissions	Use	Manage	Admin
Owner	☑	☑	☐
Group	☐	☐	☐
Other	☐	☐	☐

Ownership	
Owner	
Group	

Attributes	
HYPERVISOR	kvm

Service Level Authentication

- Global Authentication:
- Single Route Authentication

Authentication at Node.js routes can be used to configure both the above.

Tasks To be Completed

Tasks to be completed

As part of the exercise1, there are two subtasks to be completed **(in two separate APIs):**

1. **Access and send following VM information in unauthenticated API**
   1. Number of users logged in (Remember to add at least one more user and try login through it).
   2. Usernames of the user logged in. Usernames of the users are to specified in the array as shown in the exercise message structure template.
   3. Number of storage disks attached (we are here only concerned about the disks and that too **Virtual disks (vd)**)
   4. Information about the above disks (disk: size). Information of the disks are to specified in the array as shown in the exercise1Message structure template.

2. **Add a authenticated API with default authentication to username “CCS” and password as “ccs_exercise1_task2”. Here success and failure messages should also be added**

Deadline for submission: 14th November 2017 15:59 EST

Hints

1. You can use Linux shell commands to get all the above information.
2. For running the Linux shell commands inside node.js check the child_process module of node.js

Node.js Client Application Provided

Install the node and npm

- Get yourself a more recent version of Node.js by adding a PPA (personal package archive) maintained by NodeSource.

```
curl -sL https://deb.nodesource.com/setup_7.x | sudo -E bash -
```

- You can now install the Node.js package.

```
sudo apt-get install nodejs
```

The nodejs package contains the nodejs binary as well as npm, so you don't need to install npm separately.

- For some npm packages to work (such as those that require building from source), you will need to install the build-essentials package

```
sudo apt-get install build-essential
```

- Test Node: `node -v` (This should print a version number, so you'll see something like this v0.10.35)
- Test NPM: `npm -v` (This should print NPM's version number so you'll see something like this 1.4.28)

Installation of required modules

1. As part of this application some modules need to be installed. For installing them run the following command from inside the directory of application.

```
npm install
```

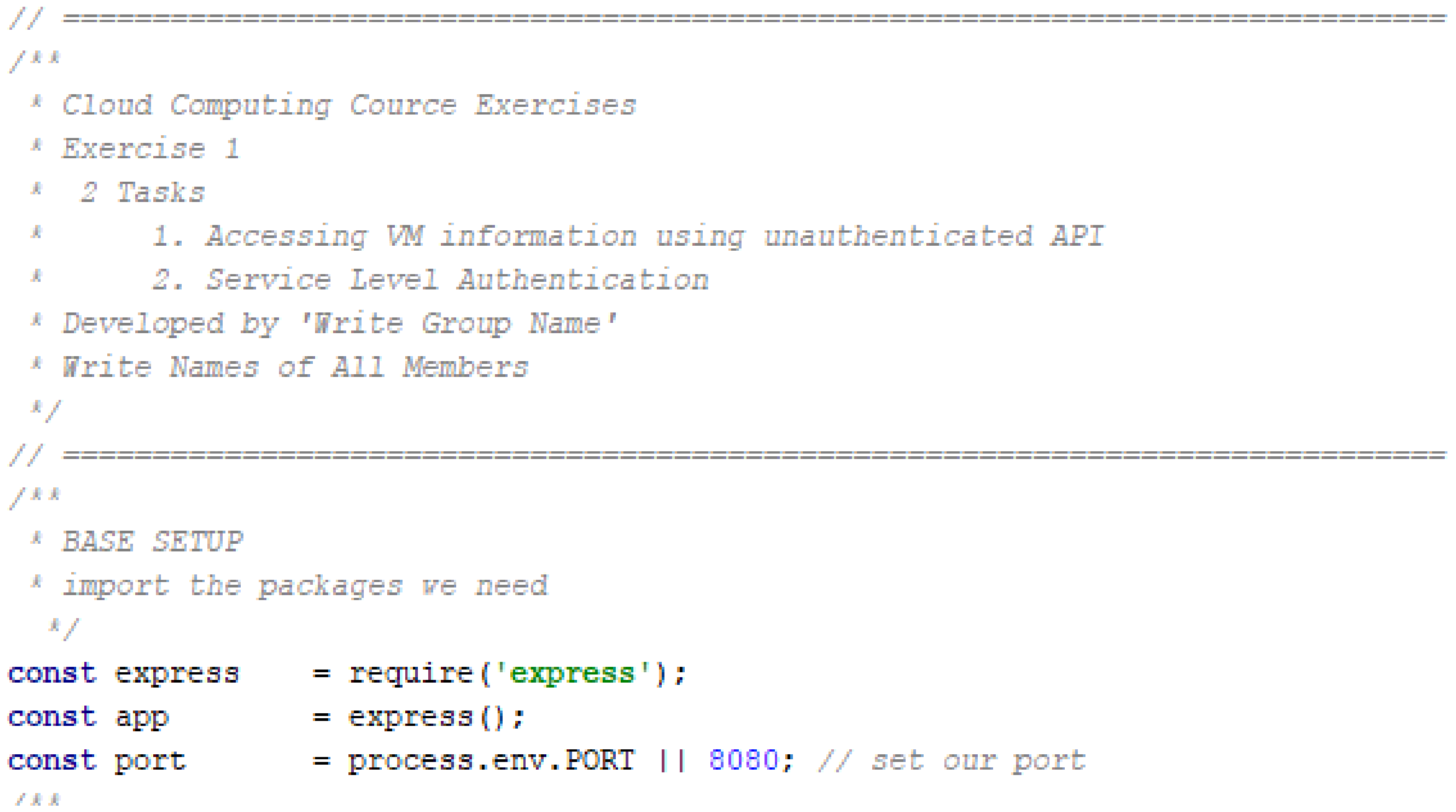
This command will install all the dependent modules mentioned in the package.json file.

2. If you need some other modules you can install them by running the command

```
npm install “module name” –save
```

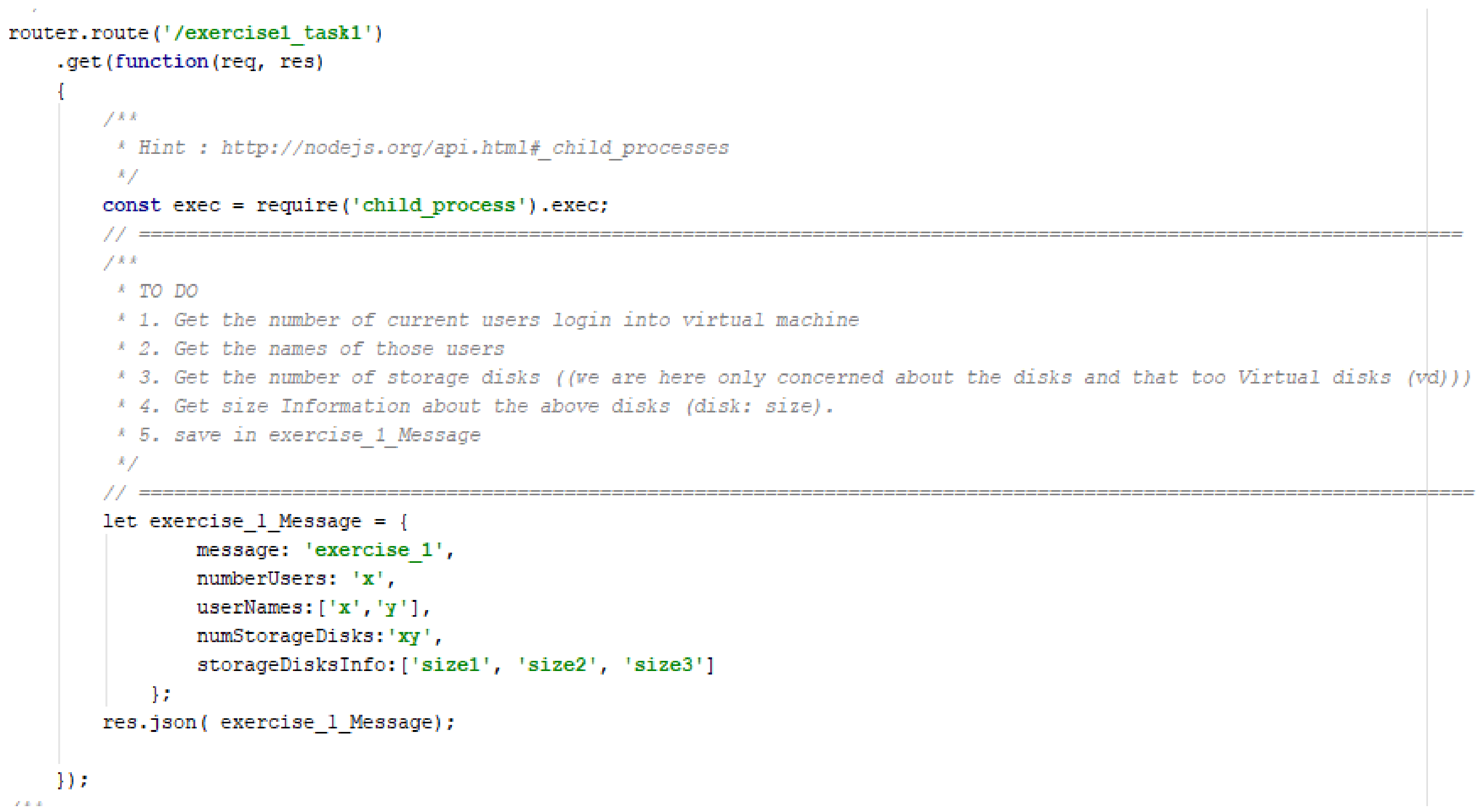
This will automatically add that module in the package.json file and now you can use it inside you development file.

Explanation of clientApplication : Base Setup



```
// ==============================================================================
/**
 * Cloud Computing Cource Exercises
 * Exercise 1
 *  2 Tasks
 *      1. Accessing VM information using unauthenticated API
 *      2. Service Level Authentication
 * Developed by 'Write Group Name'
 * Write Names of All Members
 */
// ==============================================================================
/**
 * BASE SETUP
 * import the packages we need
  */
const express     = require('express');
const app         = express();
const port        = process.env.PORT || 8080; // set our port
/**
```

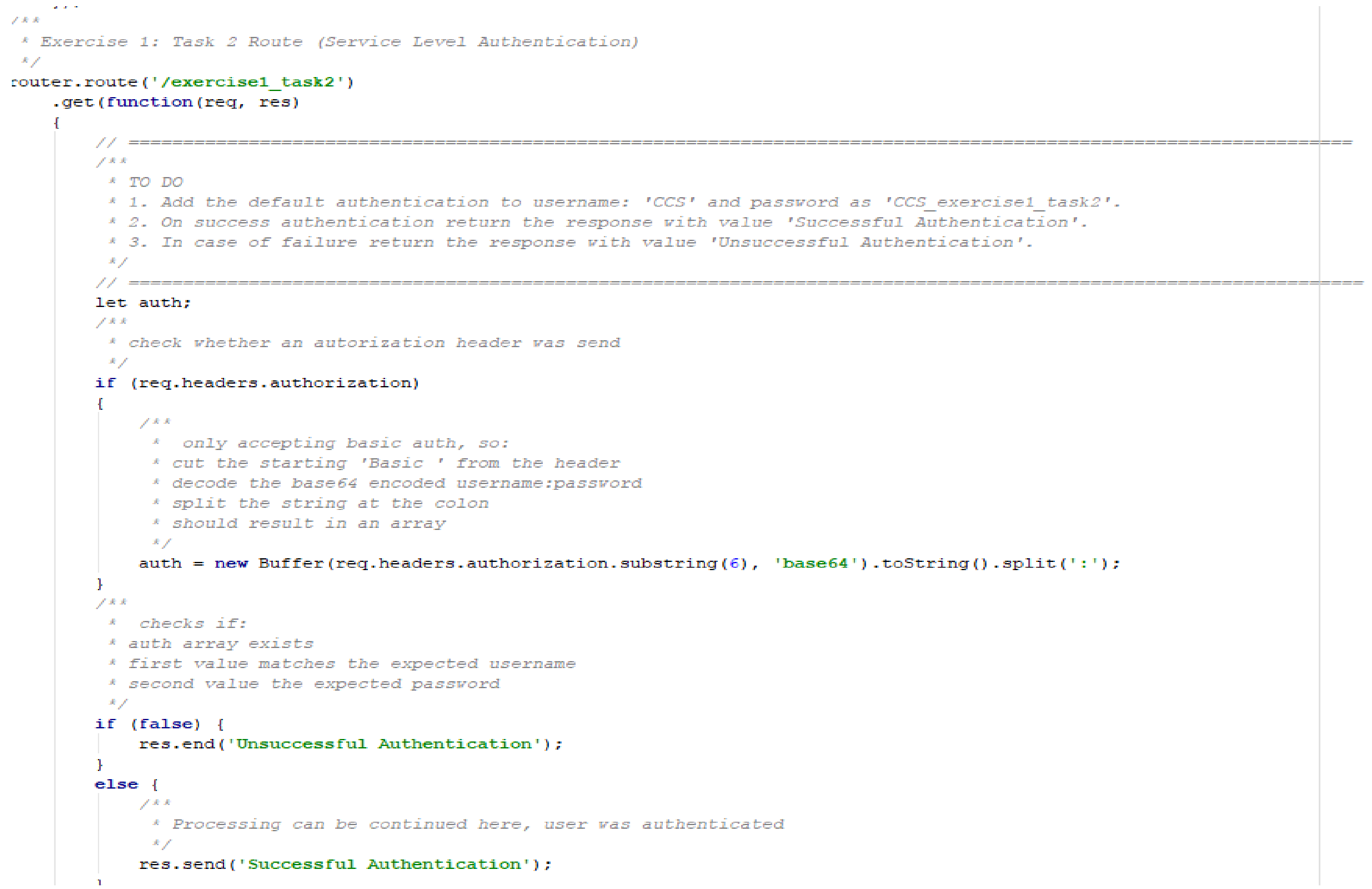
Explanation of clientApplication : Exercise1_Task1



```
router.route('/exercise1_task1')
    .get(function(req, res)
    {
        /**
         * Hint : http://nodejs.org/api.html#_child_processes
         */
        const exec = require('child_process').exec;
        // ================================================================================================
        /**
         * TO DO
         * 1. Get the number of current users login into virtual machine
         * 2. Get the names of those users
         * 3. Get the number of storage disks ((we are here only concerned about the disks and that too Virtual disks (vd)))
         * 4. Get size Information about the above disks (disk: size).
         * 5. save in exercise_1_Message
         */
        // ================================================================================================
        let exercise_1_Message = {
                message: 'exercise_1',
                numberUsers: 'x',
                userNames:['x','y'],
                numStorageDisks:'xy',
                storageDisksInfo:['size1', 'size2', 'size3']
            };
        res.json( exercise_1_Message);

    });
```

Explanation of clientApplication : Exercise1_Task2



```
/**
 * Exercise 1: Task 2 Route (Service Level Authentication)
 */
router.route('/exercise1_task2')
    .get(function(req, res)
    {
        // ==========================================================================================================
        /**
         * TO DO
         * 1. Add the default authentication to username: 'CCS' and password as 'CCS_exercise1_task2'.
         * 2. On success authentication return the response with value 'Successful Authentication'.
         * 3. In case of failure return the response with value 'Unsuccessful Authentication'.
         */
        // ==========================================================================================================
        let auth;
        /**
         * check whether an autorization header was send
         */
        if (req.headers.authorization)
        {
            /**
             *  only accepting basic auth, so:
             * cut the starting 'Basic ' from the header
             * decode the base64 encoded username:password
             * split the string at the colon
             * should result in an array
             */
            auth = new Buffer(req.headers.authorization.substring(6), 'base64').toString().split(':');
        }
        /**
         *  checks if:
         * auth array exists
         * first value matches the expected username
         * second value the expected password
         */
        if (false) {
            res.end('Unsuccessful Authentication');
        }
        else {
            /**
             * Processing can be continued here, user was authenticated
             */
            res.send('Successful Authentication');
```

Node.js Client Application Deployment

1. Now your application is ready to be deployed on VM. Run the following command to start the application:

   node clientApplication.js

   After running this, on console you will see

   “Server started and listening on port 8080”

2. Now your application is deployed on the server. Open your browser on your local machine and enter the address as

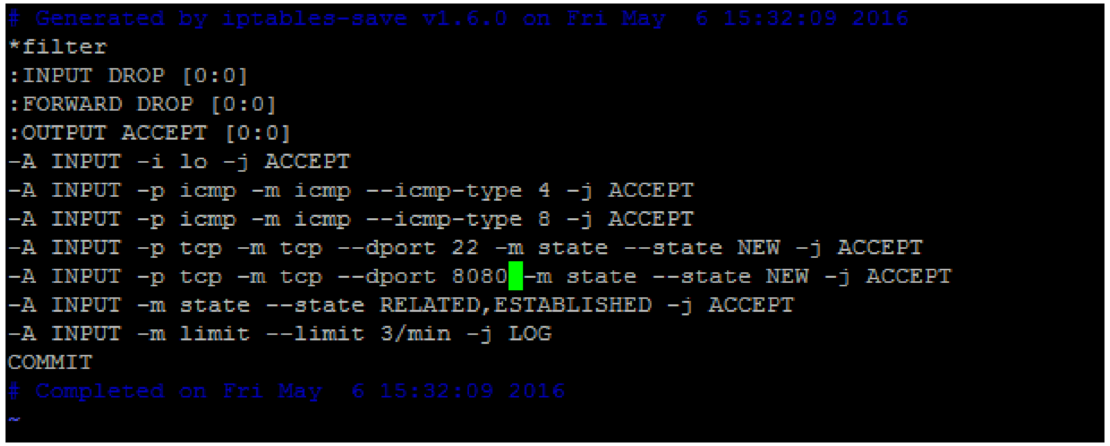
   http://IP_ADDRESS_VM:8080/exercises

```
'Welcome to Cloud Computing Exercises API
```

Node.js Client Application Deployment : Port unblock

- If your request timed out, your VM probably has some firewall rules in place prevent a user to call your web server from the outside.
- The iptables rules are located in the file **/etc/iptables/rules.v4**. Open this file with your favourite editor:
- After line 9 insert a new line allowing incoming connections on port 8080:

```
-A INPUT -p tcp -m tcp --dport 8080 -m state --state NEW -j ACCEPT
```



```
# Generated by iptables-save v1.6.0 on Fri May  6 15:32:09 2016
*filter
:INPUT DROP [0:0]
:FORWARD DROP [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -i lo -j ACCEPT
-A INPUT -p icmp -m icmp --icmp-type 4 -j ACCEPT
-A INPUT -p icmp -m icmp --icmp-type 8 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 22 -m state --state NEW -j ACCEPT
-A INPUT -p tcp -m tcp --dport 8080 -m state --state NEW -j ACCEPT
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -m limit --limit 3/min -j LOG
COMMIT
# Completed on Fri May  6 15:32:09 2016
~
```

- Save it and to apply the new iptables rules, you need to reload them to your local firewall system.

```
iptables-restore < /etc/iptables/rules.v4
```

Submission

Submission

You already have done a lot of tasks, now it's the time to get grade for those tasks. To submit your application results you need to follow this :

1. Open the cloud Class server url
2. Login with your provided username and password.
3. After logging in, you will find the button for **exercise1**
4. Click on it and a form will come up where you must provide your **VM ip** on which your application is running.
5. Then click submit.
6. You will get the result points from server (Admin decision will be the last)

Remember no cheating and no Hacking ☺

Important points to Note:

1. Make sure your VM and your application is running after following all the steps mentioned in this manual.
2. We will grade you based upon the number of tasks completed by you.
3. You will get to see, what your application has submitted to the server.
4. You can submit as many times until the deadline of exercise.
5. Multiple submission will overwrite the previous results.

Good Luck and Happy Coding☺

Thank you for your attention! ☺

Questions?