Business Services

Flexible Engine

Startup Guide

This guide presents the deployment of a web server accessible from the internet and its database, on the Flexible Engine platform

version July 2017

Legal Notice

Orange Business Services assumes no responsibility for any malfunctioning of the service and / or loss of customer data due to non-compliance with the terms of use of the service by the Customer.

Table of contents

Glossary

AZ : Availability Zone

ECS : Elastic Cloud Server

EIP : Elastic IP

FE : Flexible Engine

RDS : Relational Database Service

VPC : Virtual Private Cloud

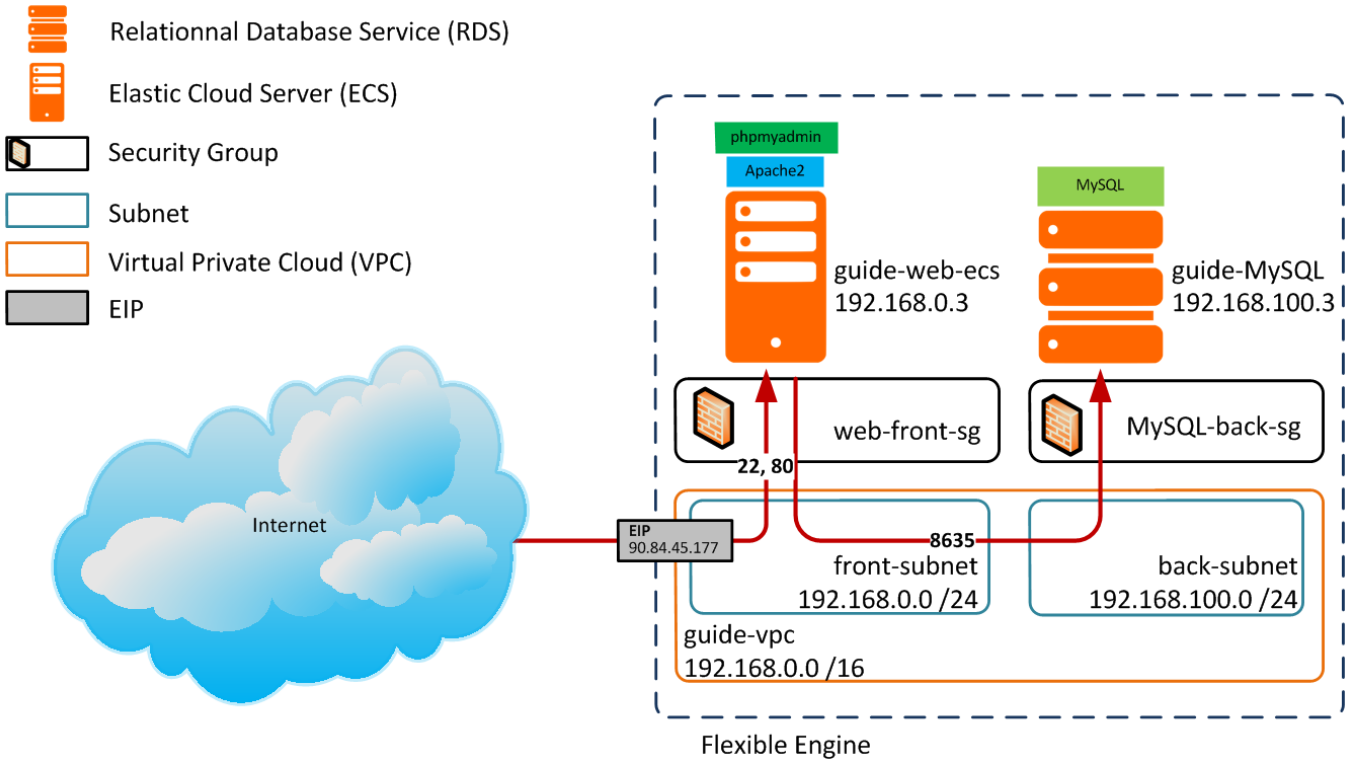
Introduction

The objective of this tutorial is to guide you step by step through the discovery of the Flexible Engine offer through the realization of a simple infrastructure composed of a web server accessible from the Internet, attached with MySQL database.

The guide starts with an empty environment, as it is provided when subscribing to the Flexible Engine offer. At the end of this guide you will have a web server accessible from internet and a MySQL database with test data. To achieve this goal, the guide describes this deployment in 10 steps:

- Step 1 : Connecting to the Flexible Engine console
- Step 2 : Creating a key pair
- Step 3 : Creating the network : Virtual Private Cloud (VPC) and Subnet
- Step 4 : Creating Security Group
- Step 5 : Creating a Relational Database Service (RDS)
- Step 6 : Creating an Elastic Cloud Server (ECS)
- Step 7 : Connecting and copying data into ECS
- Step 8 : Importing data into RDS
- Step 9 : Installing phpMyAdmin
- Step 10 : Test for proper functioning

The diagram below illustrates this target infrastructure:

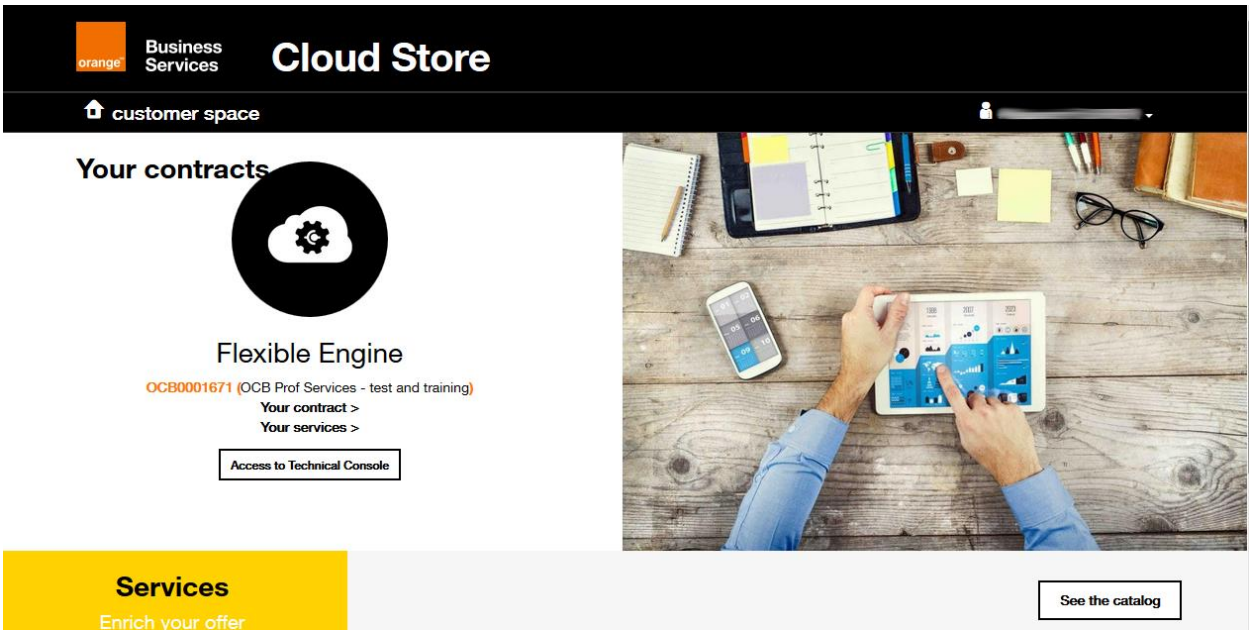


Step 1 – Connecting to the Flexible Engine Console

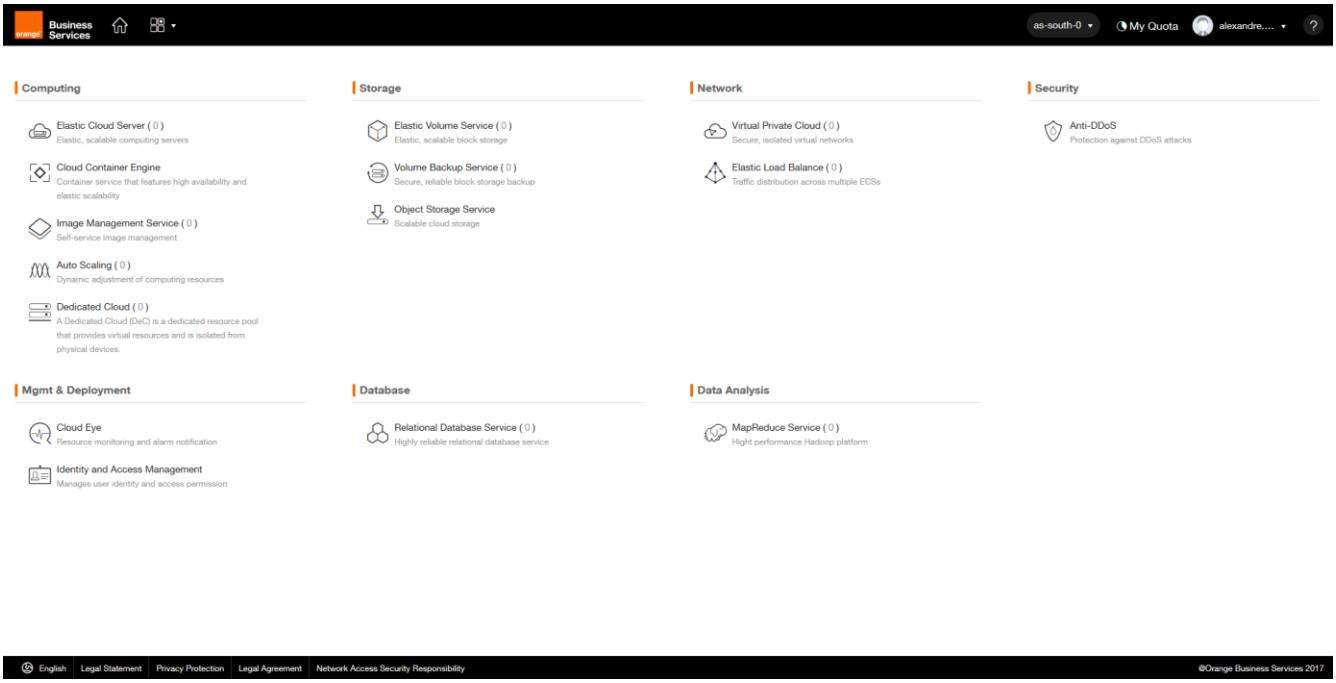
The first step of the tutorial is to show you the main view and some features of the Flexible Engine console.

You can access the console by two paths:

- http://selfcare.cloud.orange-business.com/ : the link to the cloud client space that allows you to manage all of your account information: user management, invoice access, and more. It also allows access to the console by clicking on **Access to Technical Console**.
- https://console.prod-cloud-ocb.orange-business.com/ : the direct link to the console

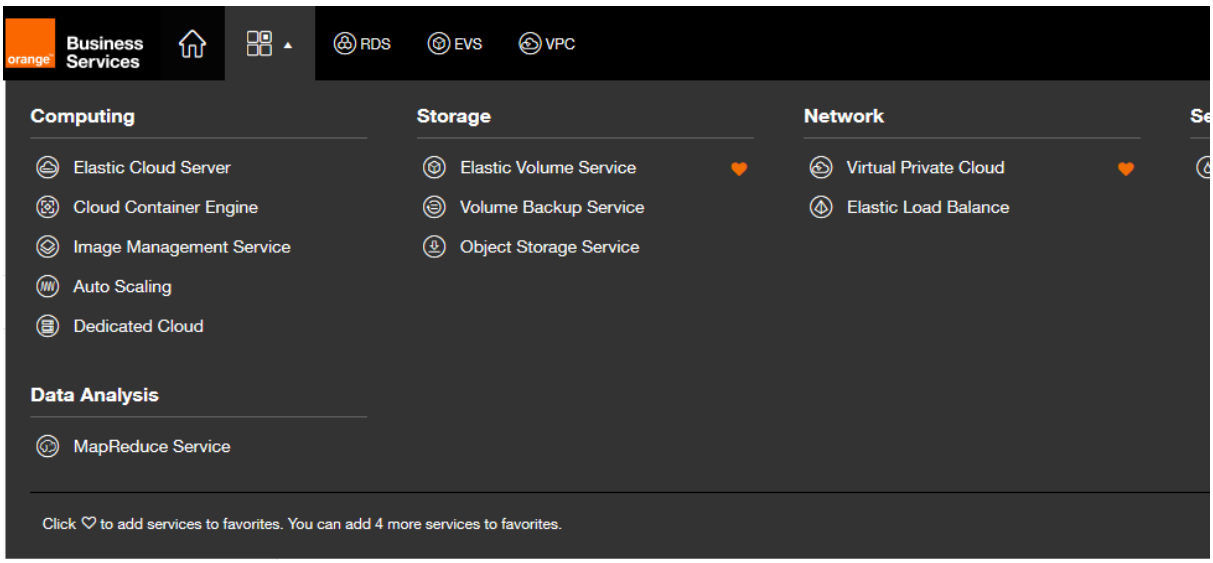


When you get on the Flexible Engine console. This page displays the main services with the number of service instances you are already consuming.



The header list contains 2 separate buttons:

- The « home » button that allows you to return to this page directly whenever you are in the console
- The second is a quick access to the various services of Flexible Engine. You can add items to favorites to customize your header slip to make it more convenient for you

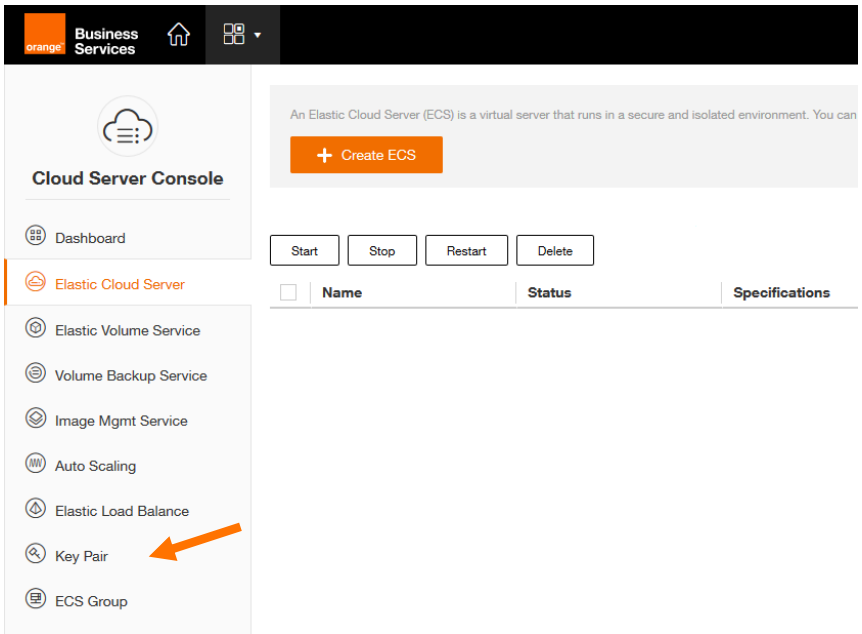


Step 2 – Creating a key pair

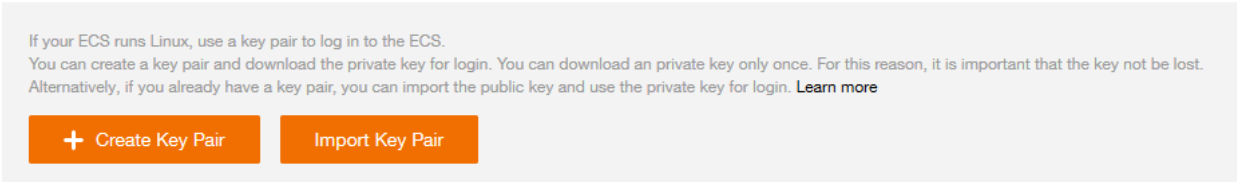
Before we start creating the components of our infrastructure, we need to create a key pair, consisting of a public key and a private key, which will allow us to securely access the servers we create:

- The public key is used, in Flexible Engine, to implement strong authentication on certain elements that you will need to deploy
- The private key will allow you to securely identify yourself on the deployed elements. Its loss would also led to loss of access for the elements deployed with the pair of keys.

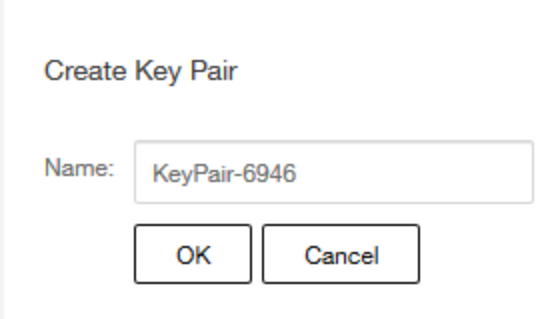
The **Key Pair** menu is a submenu of the cloud server console. To access it, first go to Elastic Cloud Server. You can now access **Key Pair** via the menu on the left.



At the top of the window, you will be proposed to create a key pair or import it.

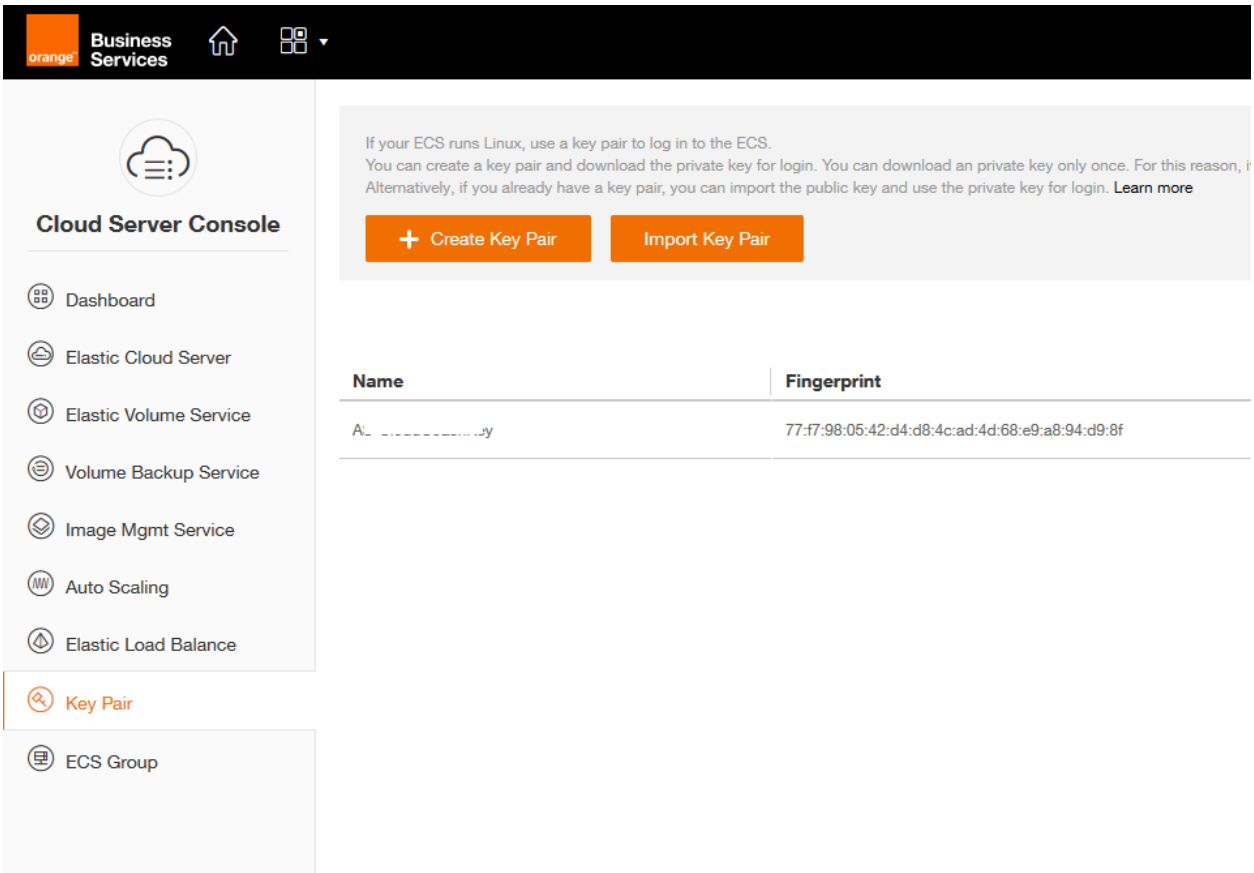


To create a key pair, click **create Key Pair**, a simple name will be requested.



By validating the creation window, you will be prompted to download a file, it is the private key. Keep this key securely as it will give access to the items you will create with it. You must be careful not to lose it.

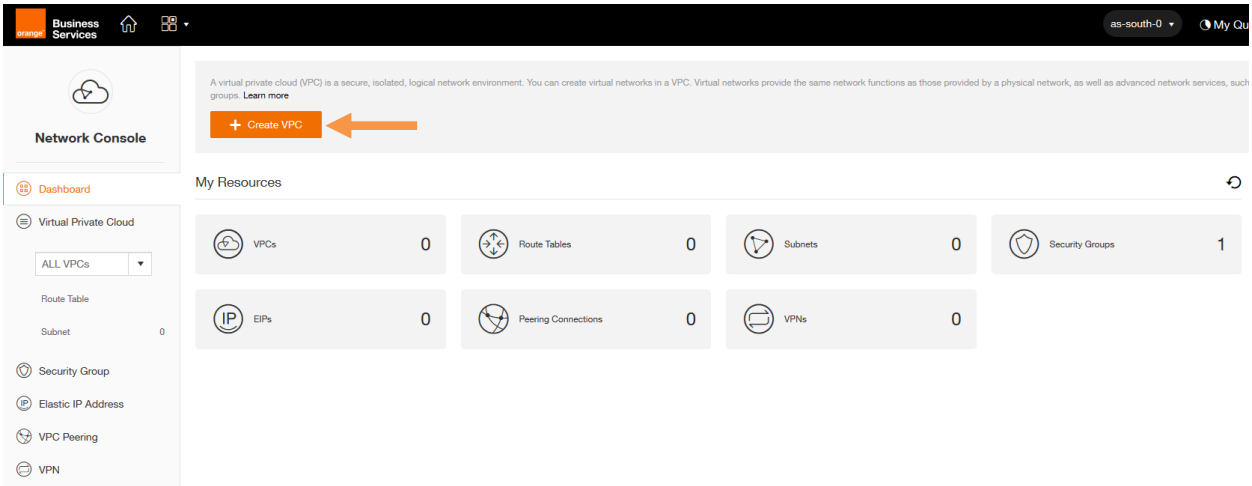
You should now see the name of your private key and its fingerprint displayed in the list.



Step 3 – Creating the network : Virtual Private Cloud (VPC) and Subnet

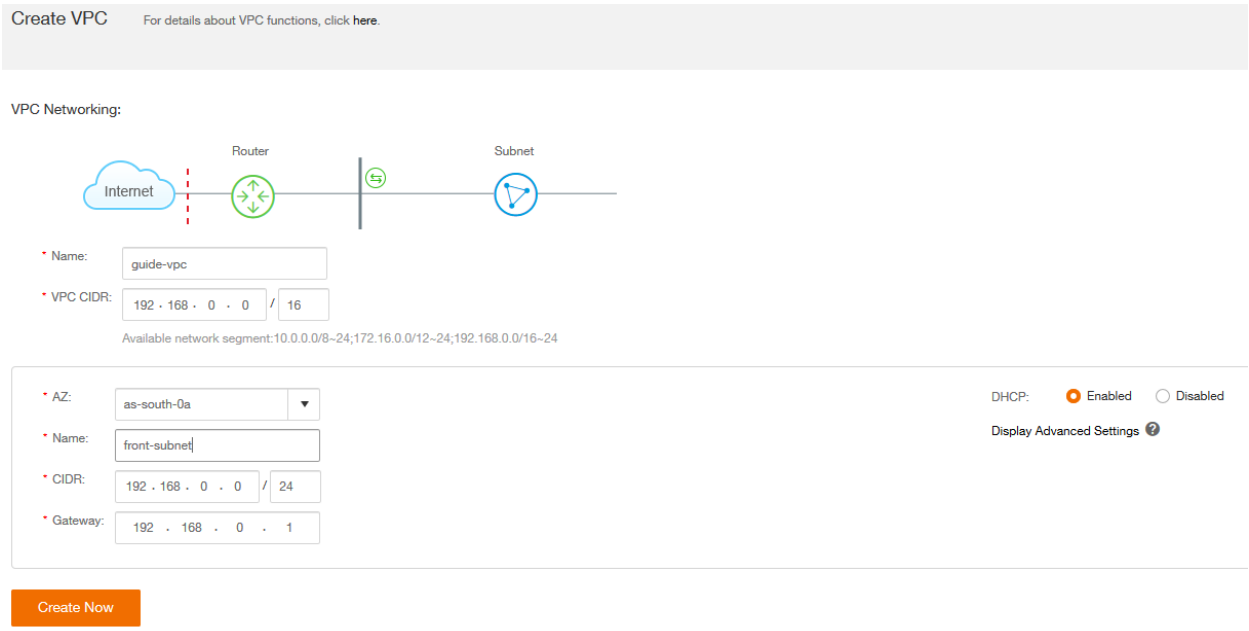
In this third step, we will create the network infrastructure in order to connect future machines.

Navigate to **Virtual Private Cloud (VPC),** and then click **Create VPC**.

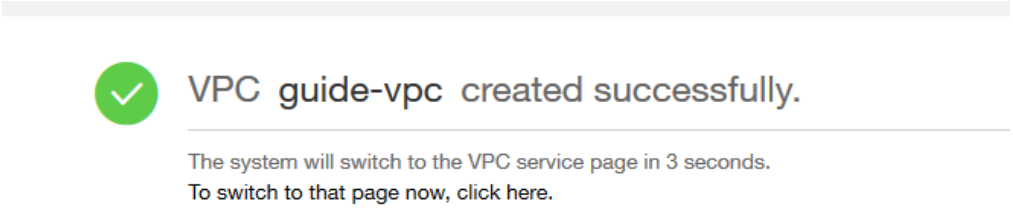


A new tab will open to give the VPC information. A VPC is an isolated virtual private network that you can configure at your convenience. A VPC has an IP range that, it then shares between its subnets. The VPCs do not have a link between them, so they can have identical addressing plans. When creating the VPC, it is also required to create the first subnet. We will indicate this information for our creation of VPC and subnet

- Name (VPC) : **guide-vpc**
- VPC CIDR : **192.168.0.0 /16**
- AZ : for Availability Zone this indicates the location of the Datacenter . Choose one and use the same to the end of the guide (exemple : « eu-west-0b » pour Datacenter B d'Europe)
- Name : **front-subnet**
- CIDR : **192.168.0.0 /24**
- Gateway : **192.168.0.1**
- DHCP : **Enabled**

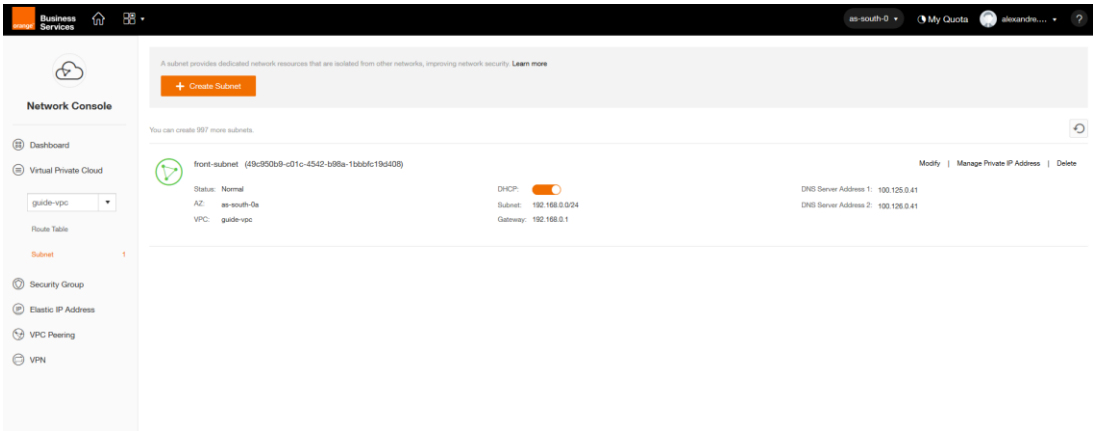


After you click **Create Now** to create this VPC and subnet, a transition page will confirm that the request is successful and will redirect you after a few seconds.



You are redirected to the page of your VPC which lists its subnets. You can view subnet information on the VPC and create more if necessary. The drop-down menu on the left allows you to browse your VPC (if you have more than one).

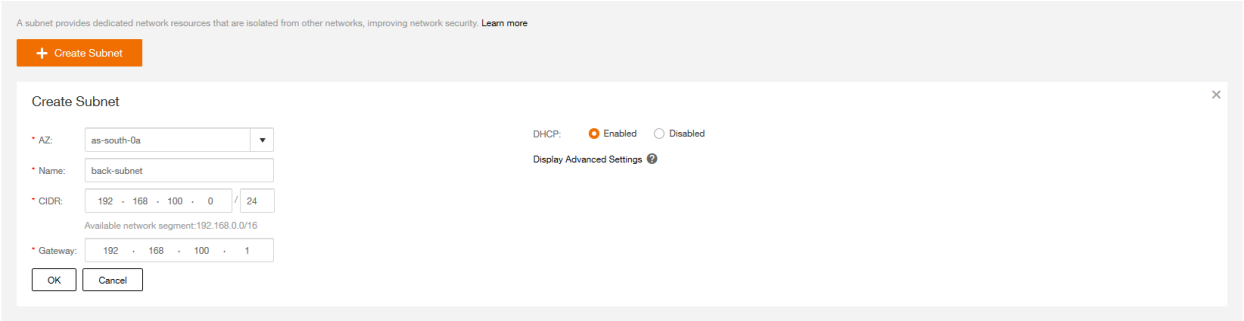
We will create the second subnet by clicking **Create Subnet**.



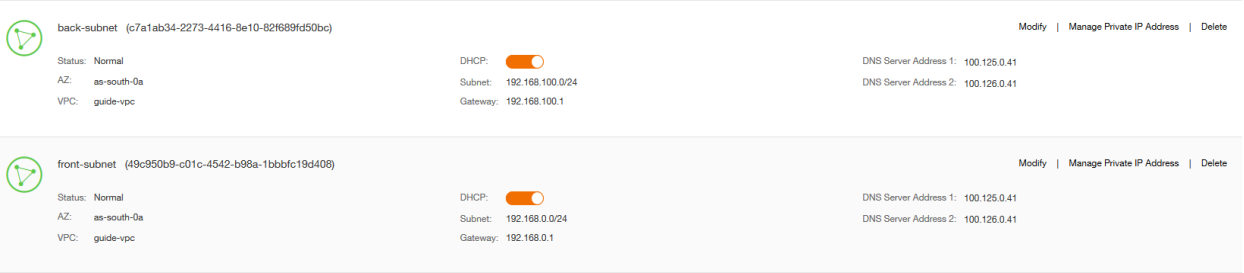
The second subnet will be for the back network :

- AZ : as previously
- Name : **back-subnet**
- CIDR : **192.168.100.0 /24**
- Gateway : **192.168.100.1**
- DHCP : **Enable**

Click on **OK** to validate



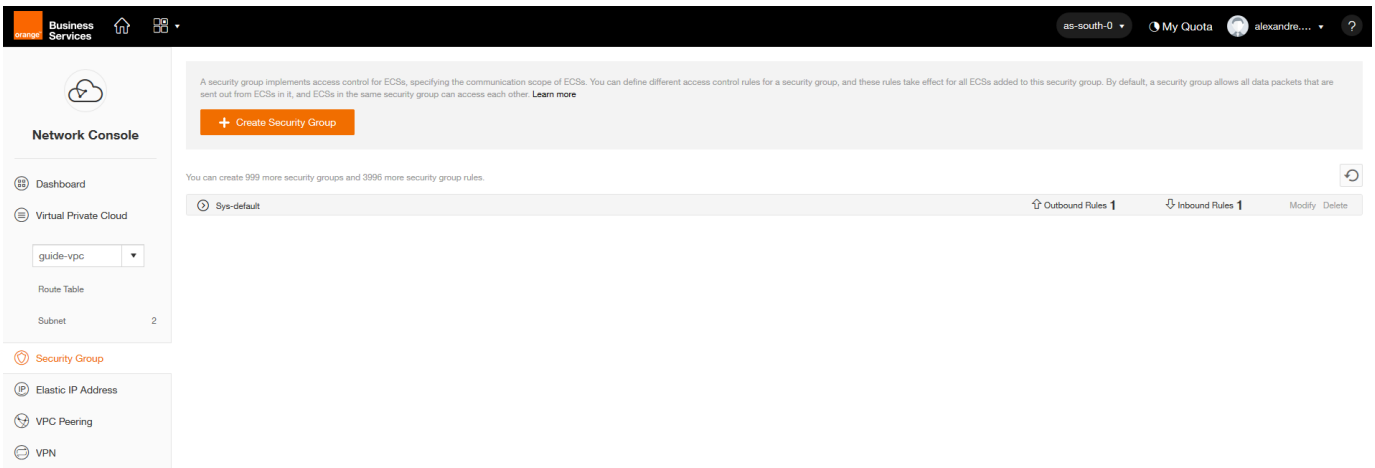
After validation, the list will show the 2 subnets of your VPC :



Step 4 – Creating Security Group

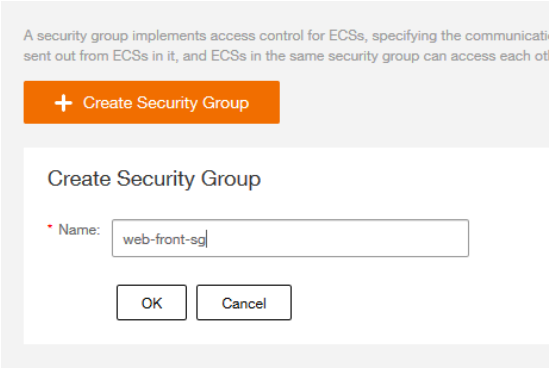
In this step we will create and configure the Security Group which will be attached to a machine. The Security Group allows the route of network flows. It can be seen as a simplified firewall.

It is accessed through the **Security Group** menu from the Network Console (Network Menu) area and Virtual Private Cloud. The screen lists the existing Security groups. For a first connection, only the default "Sys-default" rule exists.



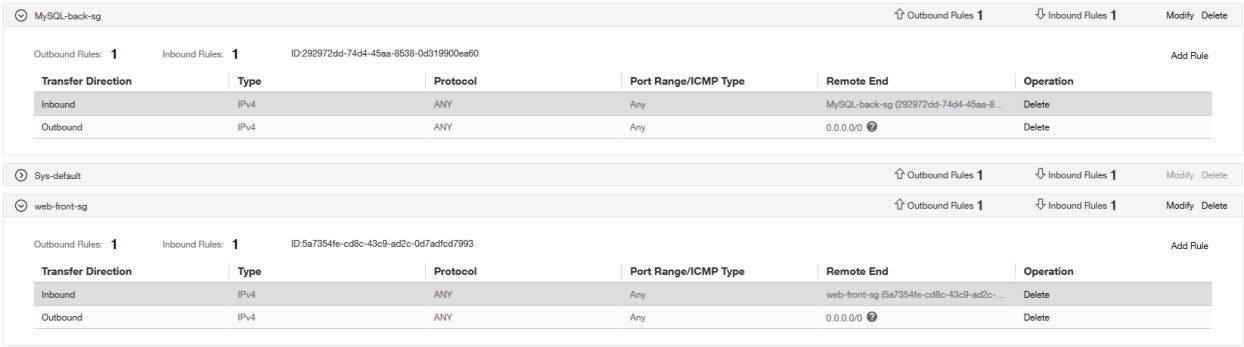
We will create 2 new Security Group for this environment to secure our machines in the back and front zone. Click **Create Security Group** at the top and simply enter a name:

- 1st : **web-front-sg**, intended for our web server
- 2nd : **MySQL-back-sg**, intended for the MySQL database



After creation, the new Security Group must be displayed in the list. You can see the details by clicking on the relevant Security group (indicated by an arrow and a name at the beginning of

the line) to display the set of rules. The two default rules will appear for each of our new Security groups.



An entry contains several parameters :

- Transfer Direction: Inbound for *incoming* and *Outbound* streams for outgoing flows
- Type: IPv4 or IPv6
- Protocol: TCP / UDP / ICMP / ANY
- Port range / ICMP Type: The port number used for TCP & UDP. It is possible to only allow certain types of ICMP such as echo, reply, etc.
- Remote End: indicates the authorized target for an *Outbound* and indicates the authorized transmitter for an *Inbound*

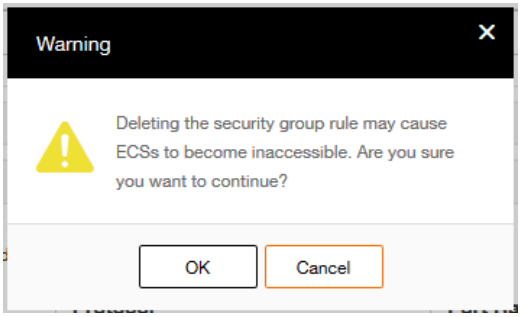
Thus the 2 routes present by default are interpreted as follows:

- Inbound / IPV4 / ANY / any / itself: allows the machines in the group to communicate with each other.
- Outbound / IPV4 / ANY / any / 0.0.0.0/0: allows the machines in the group to exit on all networks.

Not needing this Inbound rule since each machine will be alone on its subnet, we will delete it on our two Security Group (Warning! In most cases, this rule is indispensable).

By clicking **delete** at the end of the line you want to delete, you will see a warning message.

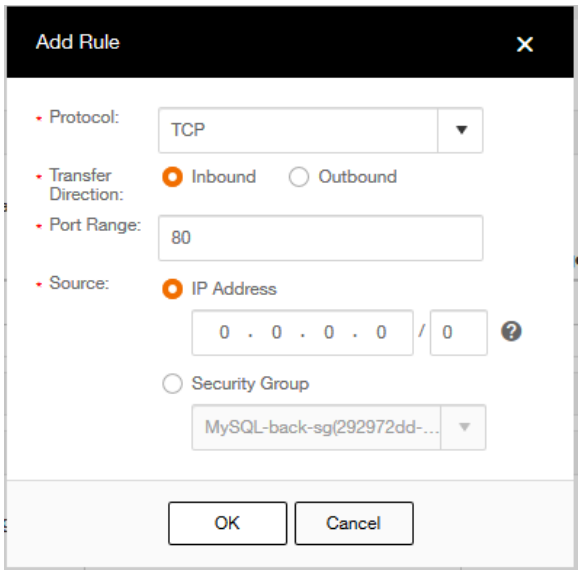
Only after validation of the pop-up by clicking **OK**, the rule will be deleted.

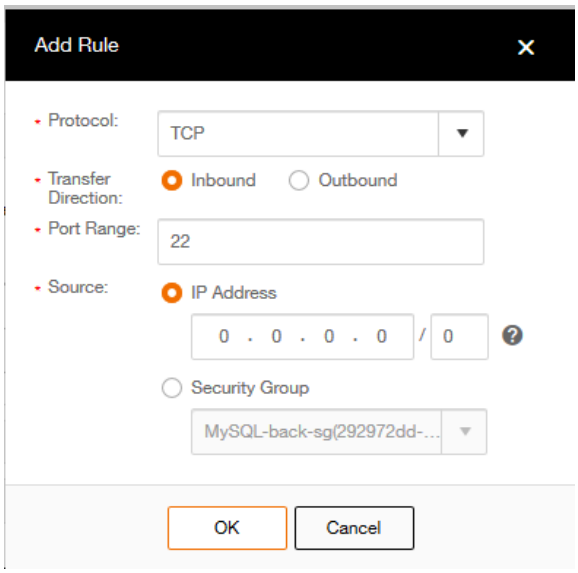


For the proper functioning of our application, it will be necessary to add several rules. We will make it accessible to all in SSH (TCP 22) and HTTP (TCP 80) to our future web server which will be in web-front-sg.

Warning ! For simplicity reasons we authorize here the connection in SSH from any Internet. In practice it is strongly recommended to open the SSH port only to authorized machines.

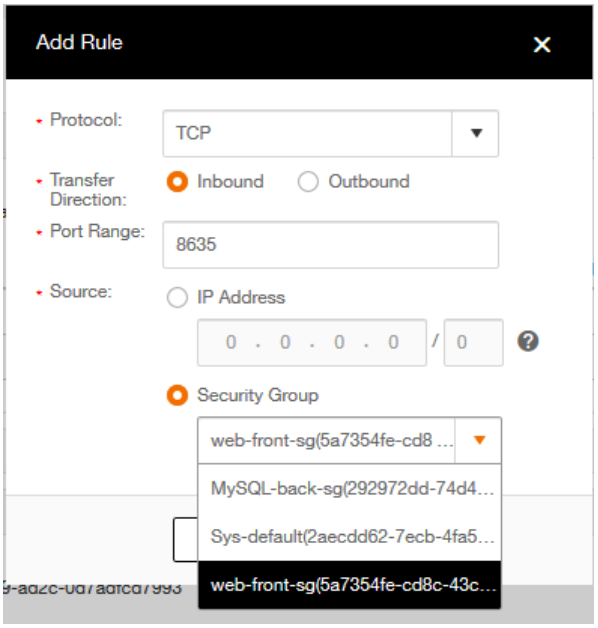
To add a rule, click **Add Rule** above the stream list of the web-front-sg Security group. Create both rules with this information:



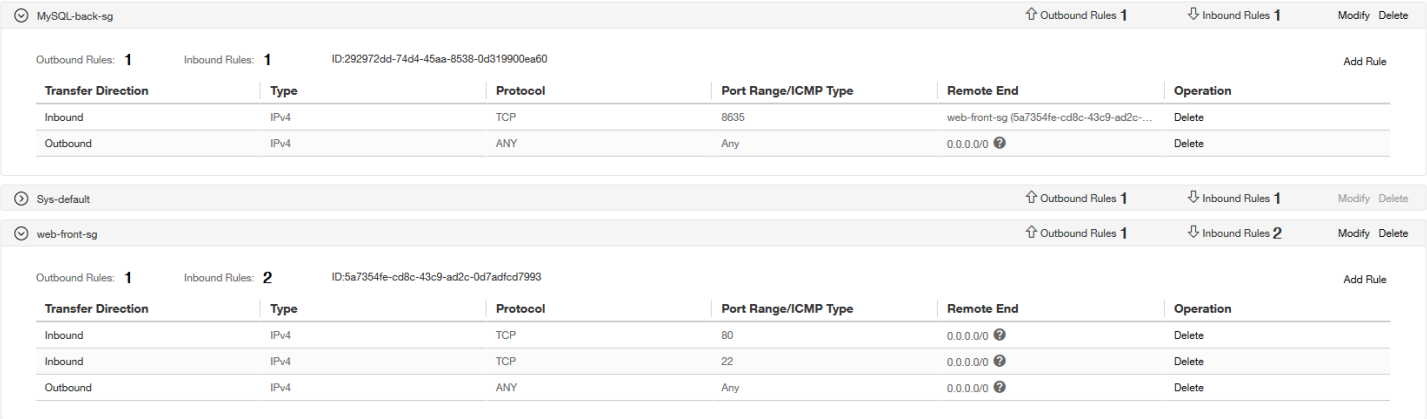


Finally, you must add a rule to allow the web server to access the MySQL database. Web-front-sg allows the output of all flows, but MySQL-back-sg must allow entry. You must add a rule to MySQL-back-sg for this:

- **TCP**
- **Inbound**
- **8635** (MySQL default port on Flexible Engine)
- Security Group : **web-front-sg** (We stay in the Flexible Engine internally, so it is possible to use the objects to increase security and avoid input errors)



After our changes, your Security Group list must be equivalent to the following :



MySQL-back-sg — Outbound Rules 1 — Inbound Rules 1 — Modify Delete

Outbound Rules: 1 Inbound Rules: 1 ID:292972dd-74d4-45aa-8538-0d319900ea60 Add Rule

Transfer Direction	Type	Protocol	Port Range/ICMP Type	Remote End	Operation
Inbound	IPv4	TCP	8635	web-front-sg (5a7354fe-cd8c-43c9-ad2c-...	Delete
Outbound	IPv4	ANY	Any	0.0.0.0/0	Delete

Sys-default — Outbound Rules 1 — Inbound Rules 1 — Modify Delete

web-front-sg — Outbound Rules 1 — Inbound Rules 2 — Modify Delete

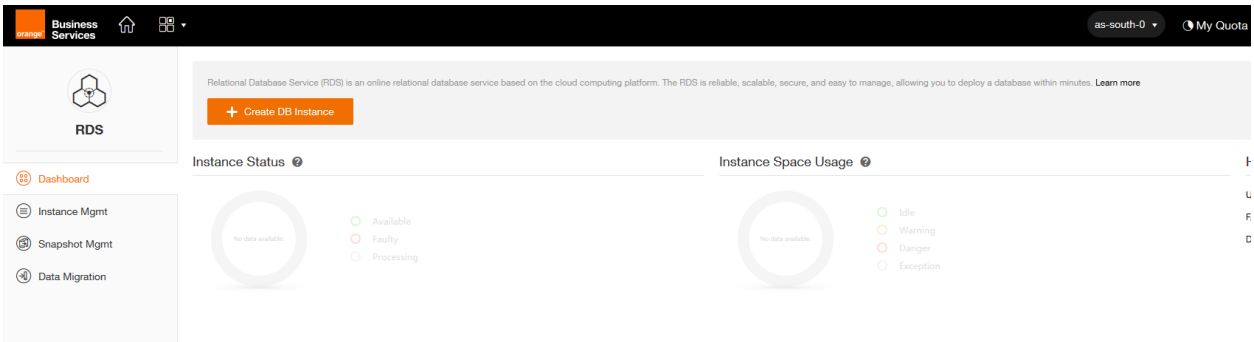
Outbound Rules: 1 Inbound Rules: 2 ID:5a7354fe-cd8c-43c9-ad2c-0d7adfcd7993 Add Rule

Transfer Direction	Type	Protocol	Port Range/ICMP Type	Remote End	Operation
Inbound	IPv4	TCP	80	0.0.0.0/0	Delete
Inbound	IPv4	TCP	22	0.0.0.0/0	Delete
Outbound	IPv4	ANY	Any	0.0.0.0/0	Delete

Step 5 – Creating a Relational Database Service

This step 5 will allow us to create a Relational Database Service (RDS). This Flexible Engine service allows you to quickly and easily deploy a relational database without worrying about the hosting machine or its operating system.

We will create a MySQL database to connect to it from the web server. To do this, go to the **Relational Database Service** menu which will redirect to Dashboard.



Click **Create DB Instance** to create an instance. A new tab will open to allow us to configure our RDS.

Instance Specifications :

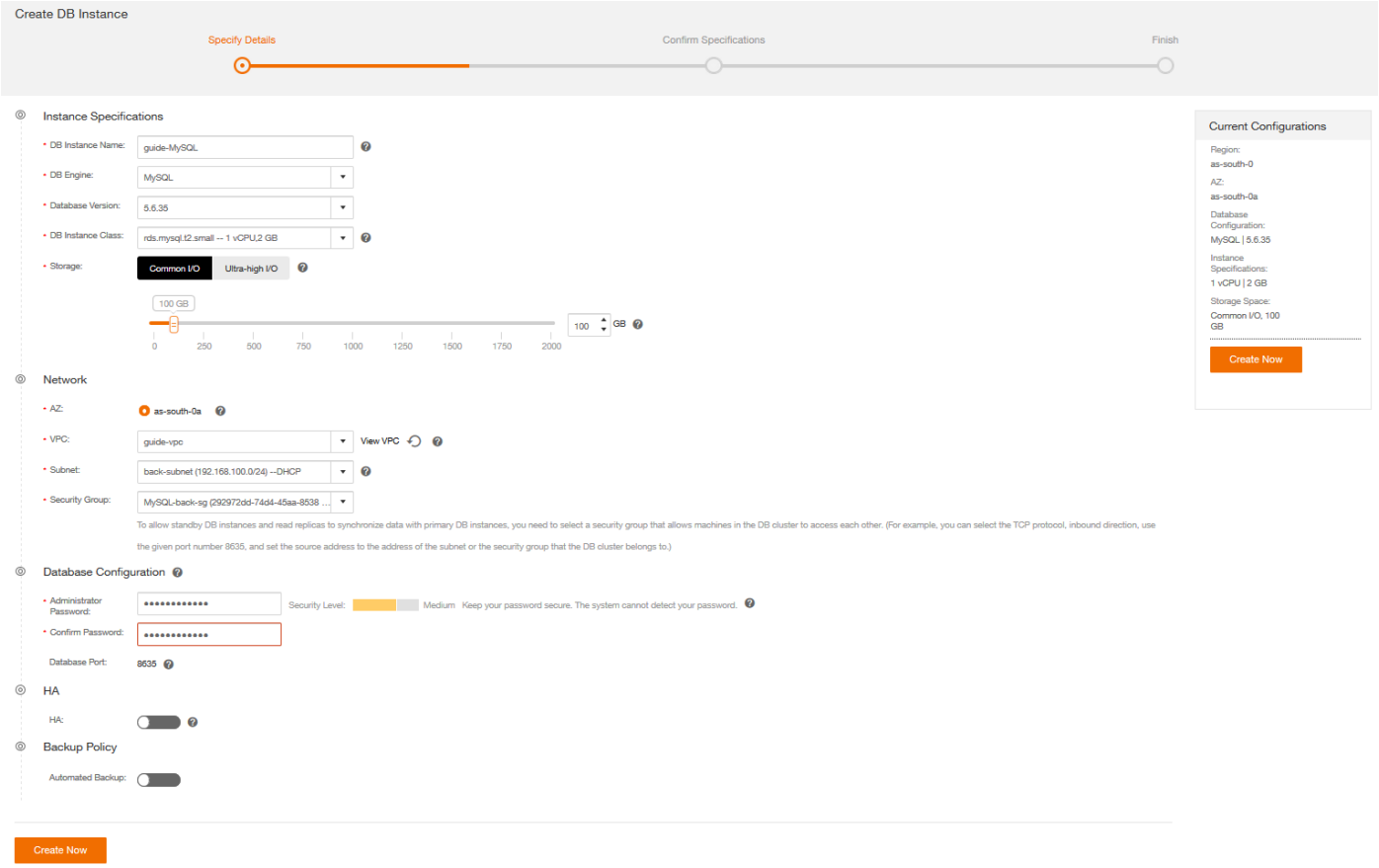
- DB instance Name : **guide-MySQL**
- DB Engine : **MySQL** (other choices are available in the list)
- Database Version : **5.6.35** (other versions are available in the list)
- DB Instance Class : **rds.mysql.t2.small – 1vCPU, 2GB** (choice according to your uses)
- Storage : Choose the Common I / O or Ultra-high I / O disk access speed and storage between 100 Go and 2000 Go
- Network
- AZ : choose the same area as VPC
- VPC : select **guide-vpc** (created in step 3)
- Subnet : select **back-subnet** (created in step 3)
- Security Group : sélectionner **MySQL-back-sg** (created in step 4)

Database Configuration

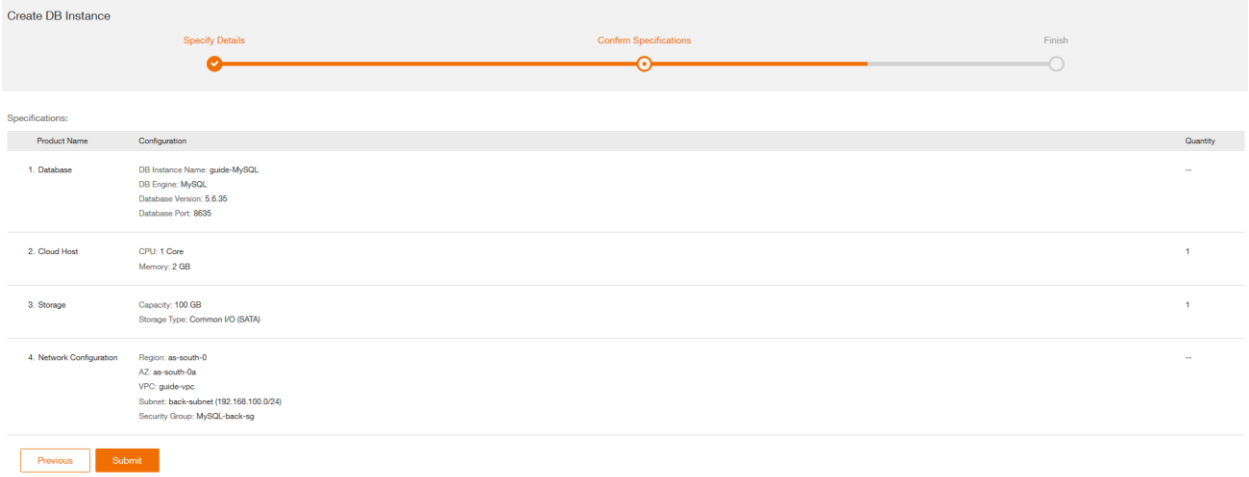
- Administrator Password : **Cloudcoach123***
- Confirm Password : **Cloudcoach123***

HA et Backup Policy : **deactivate**. These elements are discussed further in another guide

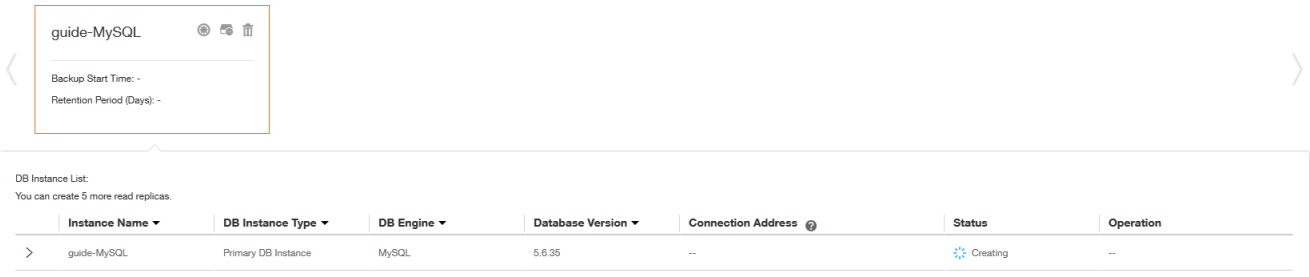
Click on **Create Now**



A summary screen is displayed. Click **Submit**, if the information is correct. Or click **Previous** to edit them.



The request to create the RDS is finished. You will be redirected for a few seconds on the page indicating the correct validation, then on the RDS instances manager, where you will find the RDS being created.

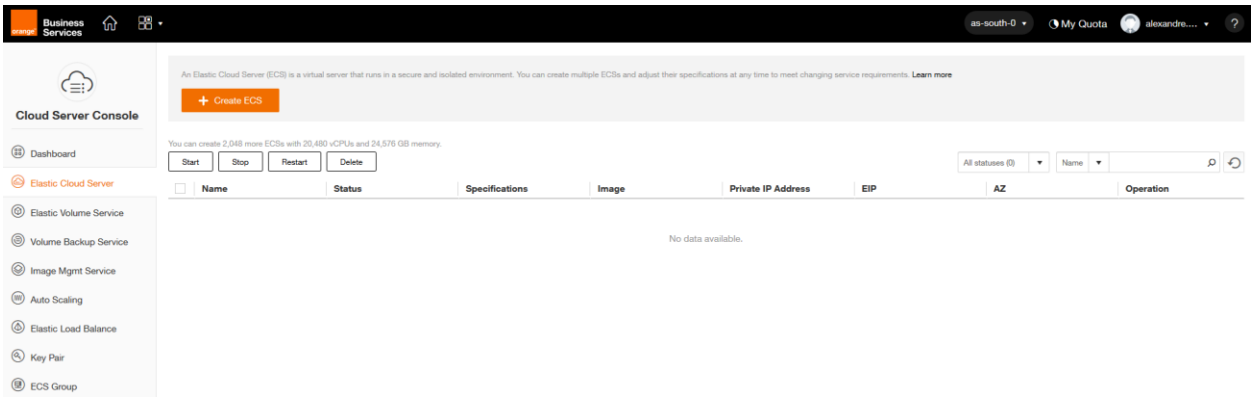


We can go to the next step, the RDS will be functional when we need to use it.

Step 6 – Creating an Elastic Cloud Server

In this step we will deploy a virtual machine. The name in Flexible Engine is Elastic Cloud Server (ECS).

We will deploy our web server. In order to do this, go to the main menu and click Elastic Cloud Server.



Click **Create ECS** to configure your ECS, a certain number of parameters have to be filled in.

Basic Information

- ECS Name : **guide-web-ecs**
- AZ : always the same
- ECS Type / vCPU /Memory : choose the number of vCPUs and RAM depending on the intended use. We are on a simple use therefore **General-purpose 1 vCPUs** and **2 GB**
- Image Type / Image : **Public image – Ubuntu**. You can choose from public images or import yours, which you will find in Private Image
- Disk : **Common I/O 40Gb** (you can choose the size, the access speed and the number of disks)

Network

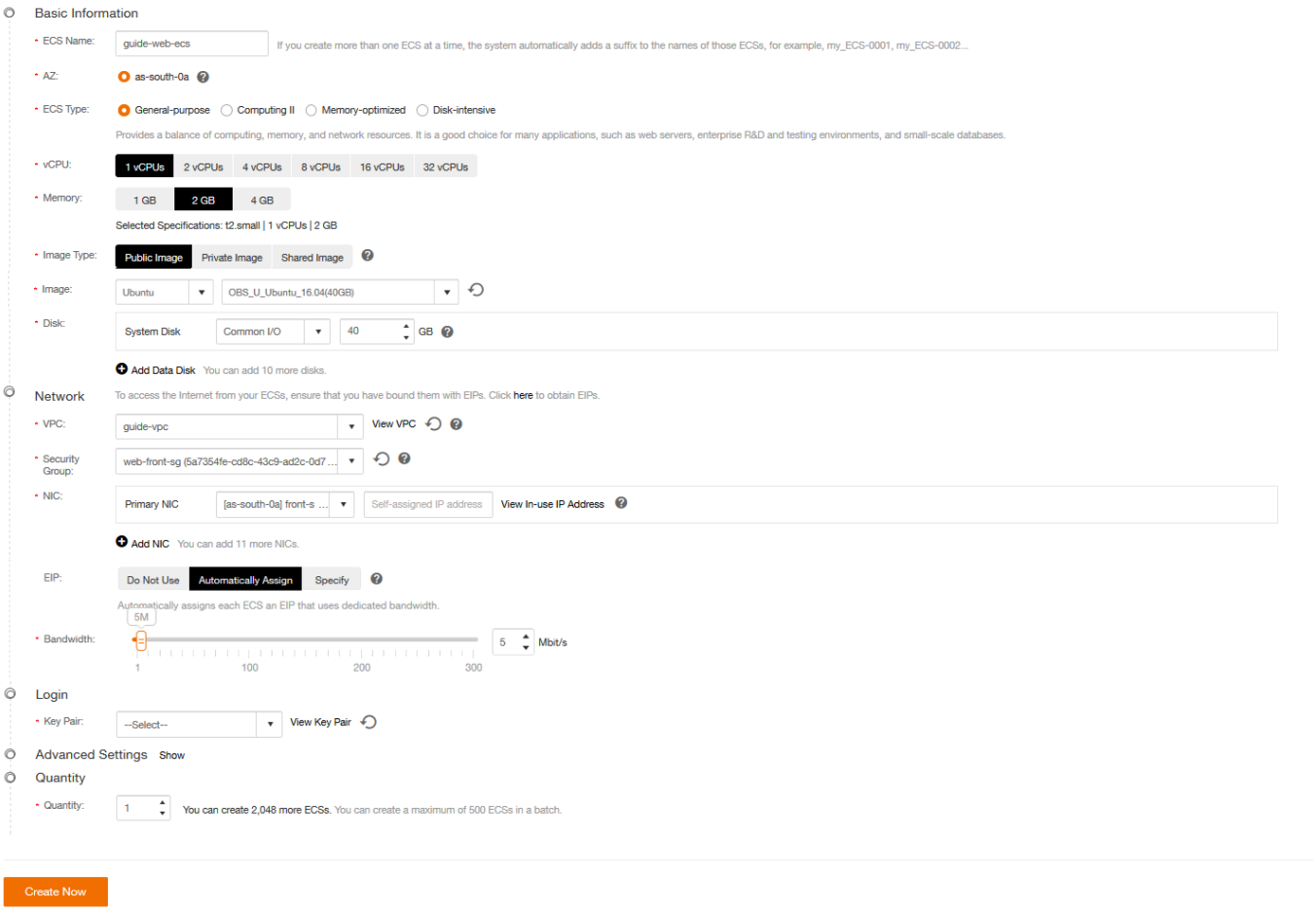
- VPC : **guide-vpc** (created in step 3)
- Security Group : **web-front-sg** (created in step 4)
- NIC : Connect the network adapter to **front-subnet** (created in step 3). You can add multiple network adapters to your ECS
- EIP : **automatically Assign** and choose a bandwidth of **5M**. This adds an Elastic IP (EIP) to the ECS to allow it to access and be accessed from the internet. What is our goal with a web server.

Login : Select the Key Pair which we created in step 2

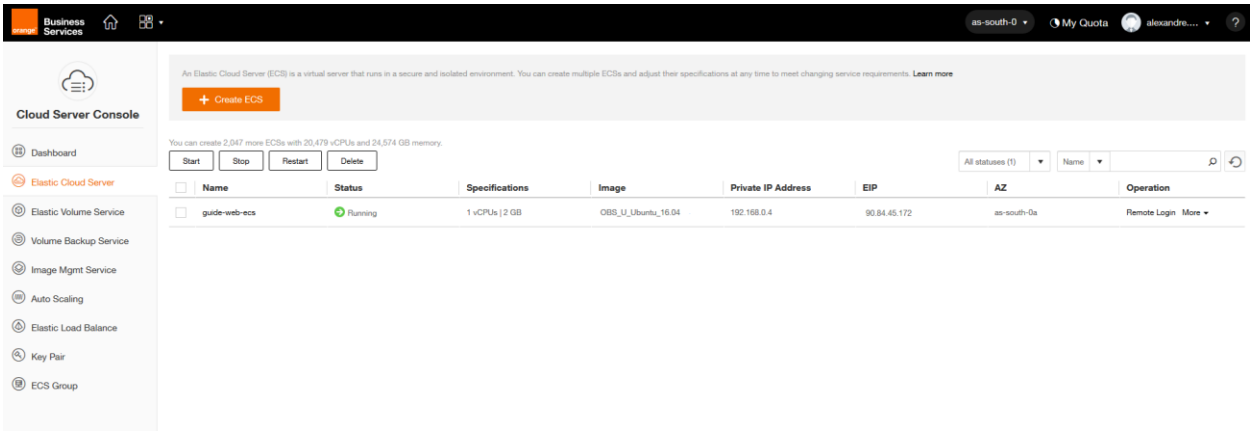
Advanced Settings: Allows you to add scripts after installation. We will not use this feature in this guide

Quantity : **1** (can create multiple machines with these same parameters)

Click on **Create Now**



After validating the first screen, you have the summary screen with the Previous and Submit buttons. By entering **Submit**, you will be redirected to the transition screen before returning to the main ECS screen.



Take note of the information in the EIP column (in our example: **90.84.45.172**). This is your public IP which will allow you to access your machine and web server thereafter.

Step 7 – Connecting and copying data to ECS

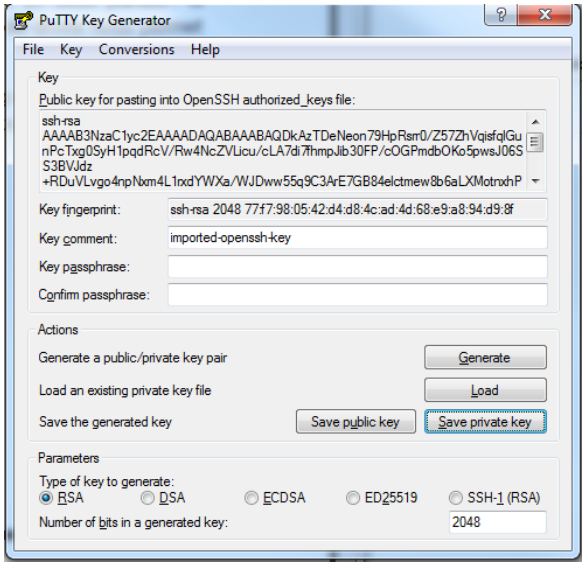
In this step 7, we will connect to the ECS in ssh and copy two files:

File name	config-db.php	importMySQL.sql
Description	Configuration file for the future phpMyAdmin	File to import into the RDS. This file creates a cloud coach database with different products and team members.
To do	Modify the line 6 : <IP server MySQL> with the IP of RDS (e.g. 192.168.100.3, but check on the RDS view)	Nothing to modify
File to import	config-db.php (see contents in annex)	importMySQL.sql (see contents in annex)

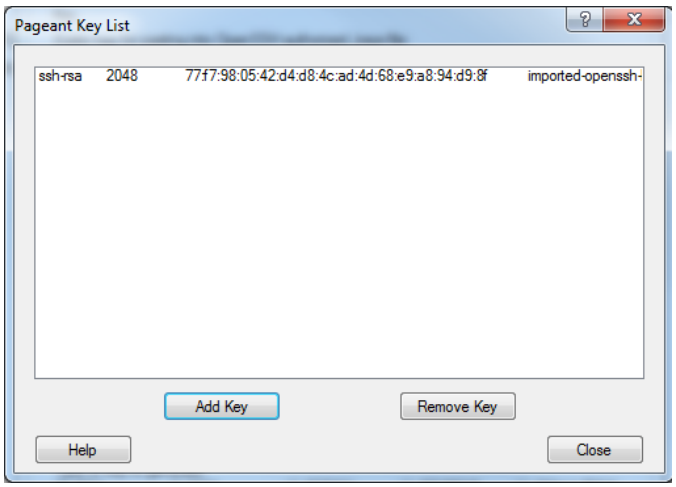
Before we can copy files, we must be able to connect to the machine. To do this, use the private key retrieved in step 2 and use the Putty program suite (www.putty.org) and the WinSCP program (https://winscp.net/eng/download.php).

You must first change the format of the private key with the help of the program puTTYgen:

- Top left: **File> Load private key** (select the .pem from step 2)
- Bottom / middle right: **Save private key**
- Choose the location, the new private key will be in ppk format



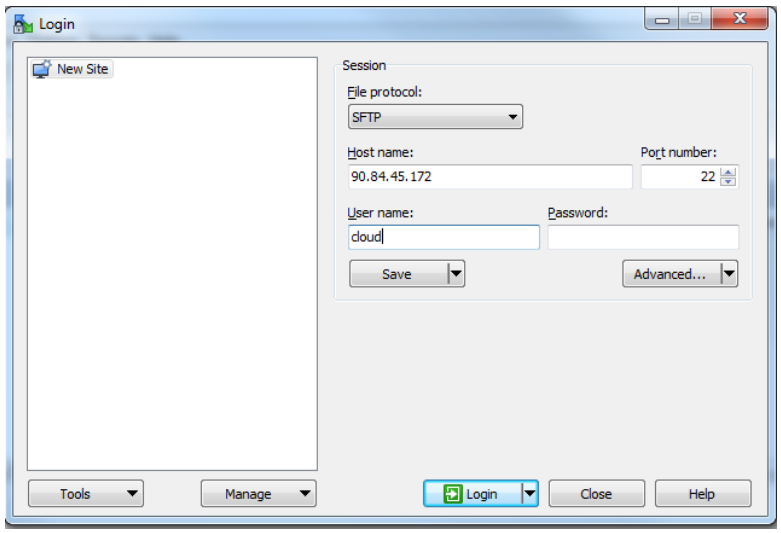
Then you have to run the **pageant program** to import the keys. This program allows to have the keys on its workstation and then to use them during the connections. Click **Add Key** to add the .ppk file you just created. After the import, you can see the fingerprint of the added key, which corresponds to the fingerprint displayed in the Key Pair view of the Flexible Engine console.



It is now possible to connect to our ECS. We will start by copying the files on the ECS with WinSCP. Log on to the server with the following information:

- File protocol **SFTP**
- Host name : IP retrieved at the end of step 6 (for my example: 90.84.45.172)
- User name : **cloud**
- Password : leave blank, the private key takes care of everything

Click **Login** (for a first login, you then have a pop-up to validate the remote host's certificate)



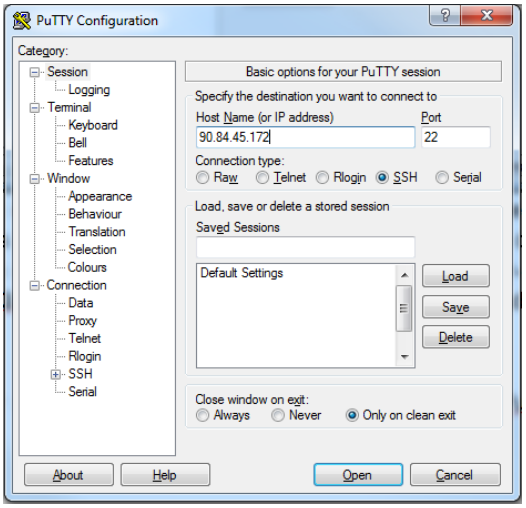
When the connection is established, WinSCP displays a window split in two parts: the left part allows us to explore our PC, while the right part allows us to explore the remote machine (our ECS). We drag and drop our local files (here importMySQL.sql and config-db.php) into the / home / cloud directory of our ECS.

When the copy is finished, you can quit WinSCP, we will now log into our console using PuTTY.

Run the Putty software and enter the login information:

- Host Name (or IP address) : **90.84.45.172**
- Port: **22**
- Click on **Open**
- (1st connection) validate the ECS certificate

Enter **cloud** when the window prompts you for a login



In order to perform these actions under Linux:

- Change the permissions on the private key and copy the files to the ECS

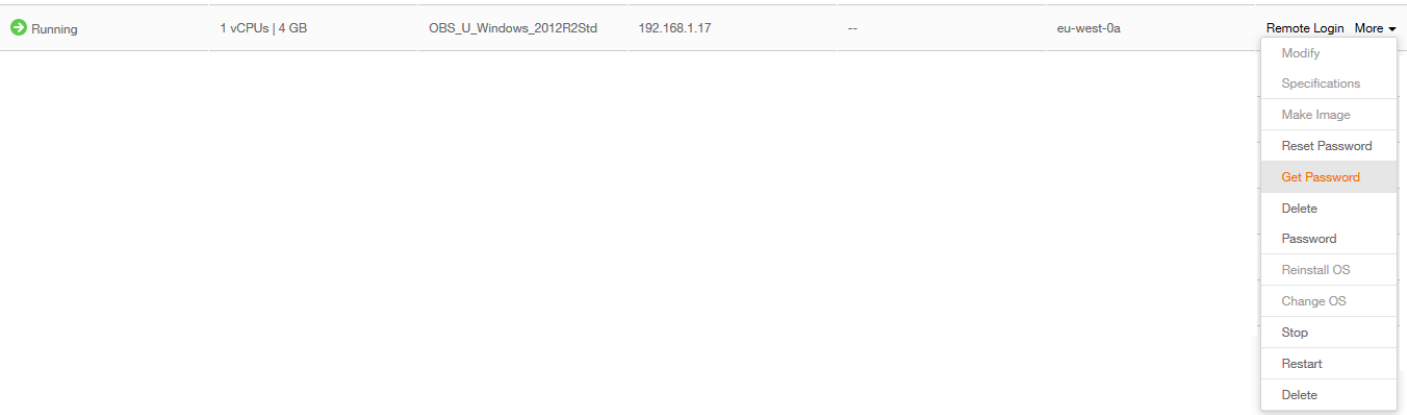
```
$ sudo chown 400 privatekey.pem

$ sudo sftp -i privateKey.pem /path/to/file/importMySQL.sql
cloud@90.84.45.172/home/cloud/ -P 22

$ sudo sftp -i privateKey.pem /path/to/file/config-db.php
cloud@90.84.45.172/home/cloud/ -P 22
```

- Connection in ssh to **ECS**

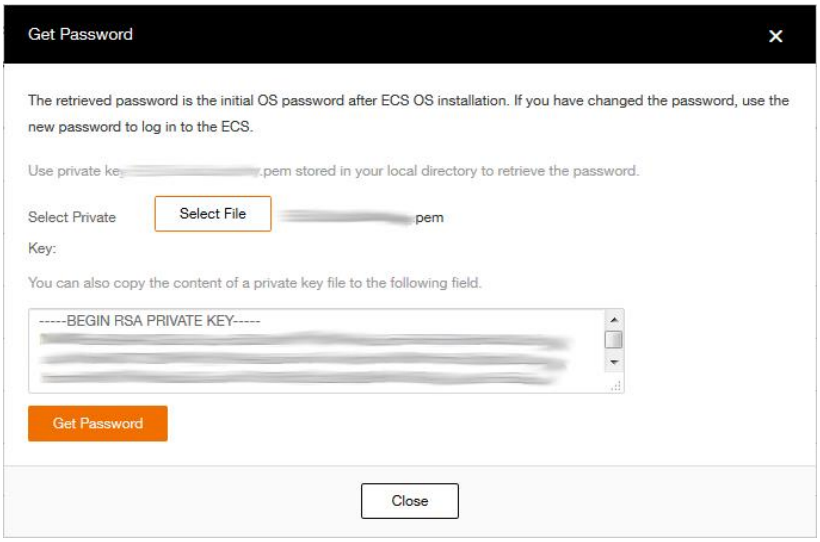
```
$ ssh -i privateKey.pem cloud@90.84.45.172
```

Small parenthesis before the next step. This guide does not deploy a machine operating under Windows, however it is interesting to know how to connect to it and to know the utility of the private key.

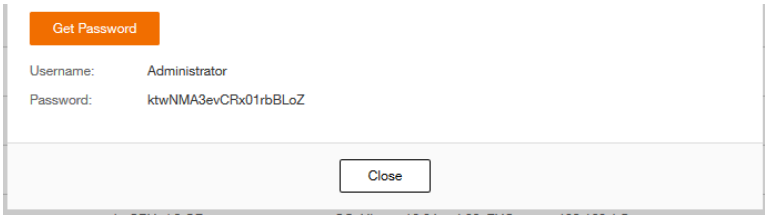
The private key allows for Windows machines to retrieve the password from the administrator account. To access the password, go to the Elastic Cloud Server page. At the end of your ECS Windows line, click on **More**. A menu opens, click **Get Password**.



A pop-up opens. Select your private key on your computer by clicking on **Select File**. The contents of the key will be displayed below it, then click **Get Password**.



The pop up will expand and display the username and password.



It is possible with these usernames to connect to the machine in RDP or by the Flexible console.

Step 8 – Importing data in RDS

In this step we will allow you to push data into your RDS MySQL database. At the end of the previous step, you are connected to the console of your ECS. We will type several commands to prepare and then import the data into the database:

- Check that the files are present

```
cloud@guide-web-ecs:~$ ls

config-db.php importMySQL.sql
```

- Update the Ubuntu repositories and install the Mysql client

```
cloud@guide-web-ecs:~$ sudo apt-get update

…

cloud@guide-web-ecs:~$ sudo apt-get install mysql-client-core-5.7

…

Do you want to continue?[Y/n/y]
```

- Import data from the .sql file, the RDS password Cloudcoach123* will be requested to validate the command

```
cloud@guide-web-ecs:~$ sudo  mysql -u root -h 192.168.100.3 -P 8635 -p <
importMySQL.sql
```

Step 9 – Installing phpMyAdmin

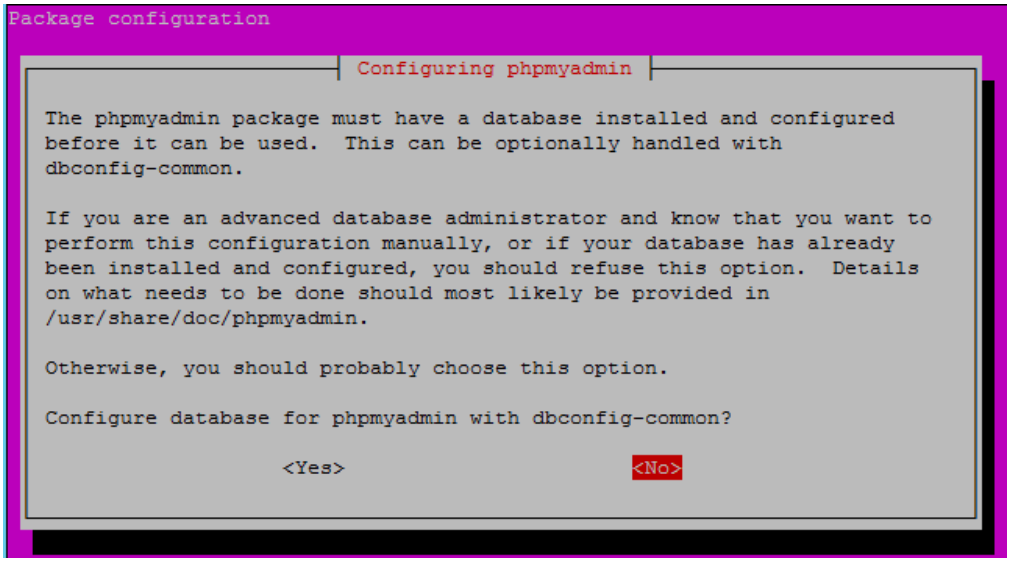
This step allows you to install phpMyAdmin and configure it using the modified (and copied) file in step 7. Here are the commands to use:

- Installing phpMyAdmin

```
cloud@guide-web-ecs:~$ sudo apt-get install phpmyadmin

…

Do you want to continue? [Y/n]y (Non dans la pop-up pendant l'installation)
```

- A windows opens, you must select apache2 as shown in the screen below (press the « space » key to select « apache2 ») and validate

- Select « no » for the "dbconfig-common" and validate



- Move the old configuration file, and copy the new one

```
cloud@guide-web-ecs:~$ sudo cp /etc/phpmyadmin/config-db.php
/etc/phpmyadmin/config-db.php.old

cloud@guide-web-ecs:~$ sudo cp /home/cloud/config-db.php /etc/phpmyadmin/config-
db.php
```

- Check that the contents of the configuration are correct

```
cloud@guide-web-ecs:~$ sudo cat /etc/phpmyadmin/config-db.php

<?php

$dbuser='root';

$dbpass='Cloudcoach123*';

$basepath='';

$dbname='phpmyadmin';

$dbserver='192.168.100.3';

$dbport='8635';

$dbtype='mysql';
```

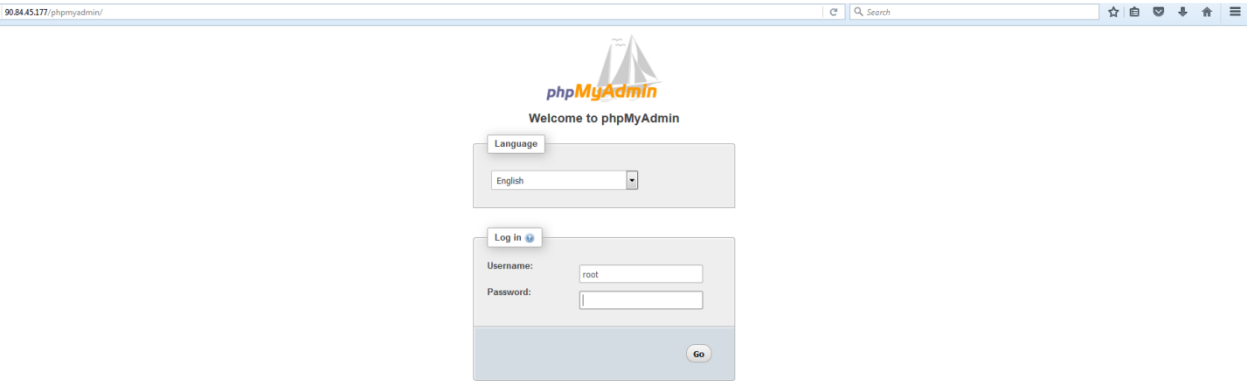
- Run the command below (by adapting the command and putting the correct IP address in it)

```
cloud@guide-web-ecs:~$ zcat/usr/share/doc/phpmyadmin/examples/create_tables.sql.gz
| sudo mysql -u root -h 192.168.100.3 -P 8635 -p
```

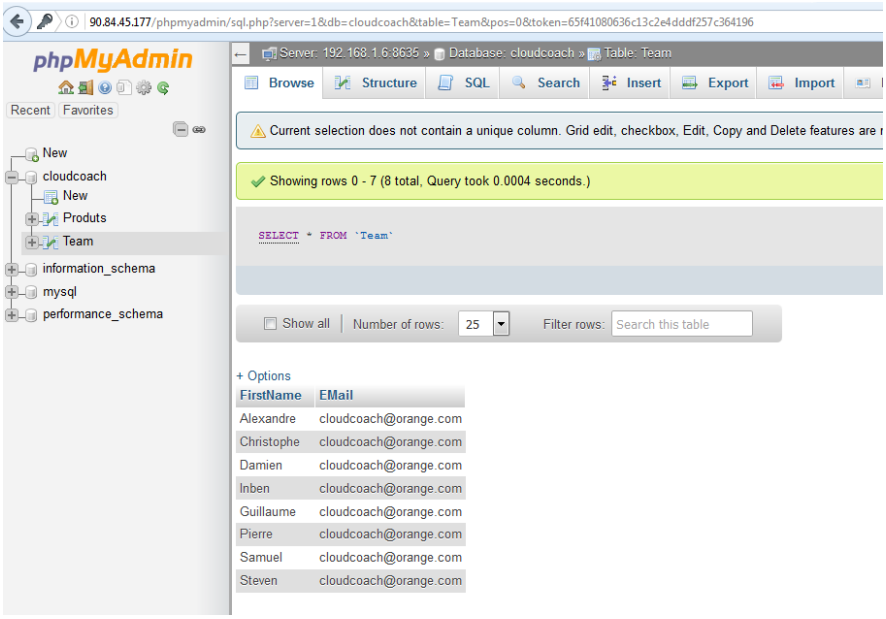
Step 10 – Test for proper functioning

This last step will allow us to test the correct operation of the installation that we have carried out throughout this guide.

- Open a web browser on your computer
- Type the address **http://<EIP-ECS>/phpmyadmin/** (For this instance <EIP-ECS> is 90.84.45.177)
- Username: **root**
- Password: **Cloudcoach123***
- Click on **Go**



After passing the login screen, you access the MySQL RDS data directly. The left side menu allows you to navigate the bases and tables. Go to the **cloudcoach** database and see the previously imported data with the sql file.



Going further

For more information on the services used in this guide, as well as on all Flexible Engine components, you can refer to the online documentation :

https://docs.prod-cloud-ocb.orange-business.com/

If you want to be accompanied in your apprenticeship, or have training adapted to your needs, you can contact Orange Cloud Coach by email at cloud.coach@orange.com.

Annex

File `config-db.php`

```
<?php
$dbuser='root';
$dbpass='Cloudcoach123*';
$basepath='';
$dbname='phpmyadmin';
$dbserver='<IP serveur MySQL>';
$dbport='8635';
$dbtype='mysql';
>
```

File `importMySQL.sql`

```
CREATE DATABASE cloudcoach;
USE cloudcoach;
CREATE TABLE Team (
    FirstName varchar(255),
    EMail varchar(255));
CREATE TABLE Product (
      ProductID int,
      Name varchar(255));
INSERT INTO Team VALUES
      ("Alexandre", "cloud.coach@orange.com"),
      ("Christophe", "cloud.coach@orange.com"),
      ("Damien", "cloud.coach@orange.com"),
      ("Guillaume", "cloud.coach@orange.com"),
      ("Inben", "cloud.coach@orange.com"),
      ("Pierre", "cloud.coach@orange.com"),
      ("Samuel", "cloud.coach@orange.com"),
      ("Steven", "cloud.coach@orange.com");
INSERT INTO Product VALUES
      (1, "Flexible Engine"),
      (2, "Flexible Computing Express"),
      (3, "Flexible Computing Premium"),
      (4, "Flexible Computing Advanced"),
      (5, "Flexible Storage");
```