Cloud Computing Applications and Services

Case-study application: Swap

2023

Swap

Consider the Swap application, used to handle class enrolment and shift exchanges. It is available from: `https://github.com/Hackathonners/swap`. The goal is to install Swap along with its dependencies and a MySQL database in separate virtual machines.

Tasks

1. Use the Vagrantfile (see warm-up guide 0) to create two VMs (*e.g.*, *server1* and *server2*). Then connect to the VMs through the SSH protocol.
2. Install and configure MySQL in *server2*:

   ```
   sudo apt install mysql-server
   ```

3. Use the MySQL client command line to:
   (a) create a database:

   ```
   sudo mysql

   CREATE DATABASE swap;
   ```

   (b) create/grant privileges to a user on *server1* to access the database:

   ```
   CREATE USER '<user>'@'<IP>' IDENTIFIED BY '<password>';

   GRANT ALL PRIVILEGES ON swap.* TO '<user>'@'<IP>' WITH GRANT OPTION;
   ```

   Note: the `<user>`, `<password>` and `<IP>` fields should be replaced by the appropriate values. For example, the `<IP>` field should correspond to the private IP address defined for *server1* (*e.g., 192.168.56.101*). The `<user>` and `<password>` fields can be chosen as desired.
   (c) exit the MySQL client console and edit the *bind-address* configuration at (`sudo` required):

   ```
   /etc/mysql/mysql.conf.d/mysqld.cnf
   ```

   Note: the *bind-address* is the IP address of the VM where the MySQL server is deployed (e.g., *server2* - 192.168.56.102).
   (d) restart MySQL service:

   ```
   sudo /etc/init.d/mysql restart
   ```

4. In the other VM (*server1*) let us install the Swap platform and its dependencies. In more detail, start by installing PHP (v7.4), as required by the application, by using the following commands:
   (a) `sudo add-apt-repository ppa:ondrej/php`
   (b) `sudo apt update`

(c) `sudo apt install php7.4 \`
`php7.4-{fpm,zIP,mbstring,tokenizer,mysql,gd,xml,bcmath,intl,curl}`

5. Install remaining dependencies (NodeJS, Composer and npm):

   (a) `sudo apt install nodejs`

   (b) `sudo apt install composer`

   (c) `sudo apt install npm`

6. Clone Swap's git repository and move to Swap directory. Now let us install and configure the Swap application:

   (a) install required packages with composer:

   `composer install`

   (b) use npm instead of yarn to install Swap:

   `npm install`

   (c) do not forget to change the database configurations (`DB_HOST`, `DB_DATABASE`, `DB_USERNAME` and `DB_PASSWORD`) at the `'.env.example'` and rename the file to `'.env'`. These should match the configurations previously defined at the Step 3 of this guide.

   (d) generate the appplication's key with:

   `php artisan key:generate`

   (e) run database migrations with:

   `php artisan migrate`

   (f) seed the database with:

   `php artisan db:seed`

7. Start Swap with:

   `php artisan serve --host=0.0.0.0`

   Note: Understand the difference between using a specific IP address versus the `0.0.0.0` IP

8. Try it out!

   (a) access Swap from your browser. The URL should contain the private IP address of *server1* and port 8000 (e.g., 192.168.56.101:8000)

   (b) log in as administrator. The username is "contact@hackathonners.org" and the password is "123456".

Extras

1. Setup an external mail server account (e.g., by using Mailtrap).
2. Use Redis for session management.

Learning outcomes Experiment the manual distributed deployment and configuration of multi-tier applications.