System Deployment and Benchmarking

Case-study application: Swap

October 16, 2020

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. Change the IP of the template VM (Warmup exercise) to 10.0.0.101.
2. Clone the VM twice (VM1 and VM2) and change the IP of VM2 to 10.0.0.102.
3. Install and configure MySQL (package mysql-server) in a virtual machine (VM2).
4. Use the mysql client command line to:
   (a) create a database.

   ```
   (sudo) mysql -p

   CREATE DATABASE swap;
   ```

   (b) create/grant privileges to a user on the other VM to access the database.

   ```
   CREATE USER 'user'@'10.0.0.101' IDENTIFIED BY 'password';

   GRANT ALL PRIVILEGES ON swap.* TO 'user'@'10.0.0.101' WITH GRANT OPTION;
   ```

   (c) edit bind-address configuration at:

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

   (d) Note: ip is the address of the database client machine, while the bind-address is the ip of the MySQL server.
   (e) restart mysql service.
5. In the other virtual machine (VM1) install the Swap platform and dependencies.
6. Install PHP, as required by the application, by using the following commands:
   (a) `sudo add-apt-repository ppa:ondrej/php`
   (b) `sudo apt-get update`
   (c) install php extensions with apt-get

   ```
   php7.4 php7.4-{fpm,zip,mbstring,tokenizer,mysql,gd,xml,bcmath,intl,curl}
   ```
7. Install remaining dependencies (NodeJS, Composer and npm)
   (a) `sudo apt-get install nodejs`
   (b) `sudo apt-get install composer`

(c) `sudo apt-get install npm`

8. Clone Swap's git repository and cd to Swap directory.
   (a) Do not forget to change the database configurations at the .env.example and rename the file to .env
   (b) `composer install`
   (c) use npm instead of yarn to install Swap:

   ```
   npm install
   ```

   (d) `php artisan key:generate`
9. Start swap with:

   ```
   php artisan serve --host=0.0.0.0
   ```

10. Try it out!

Extras

1. Setup an external mail server account (mailtrap).
2. Use Redis for session management.

Questions

1. What is this application's architecture and what pattern(s) are present?
2. What would you expect the bottleneck of this application to be? Why?
3. How would you scale this application? Which patterns would you use? Why?
4. How would you benchmark this application?

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