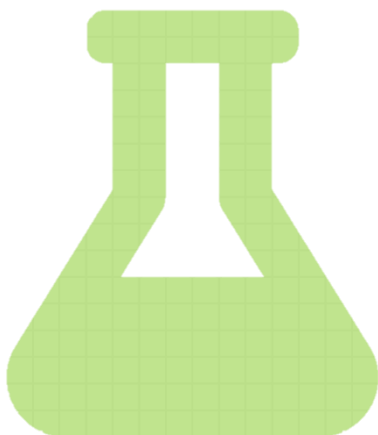




*Turbo
Consulting*



Turbo Consent Deployment Instructions April 2018



Contents

1 Introduction2

2 Clone the Git Repository2

3 MySQL Database Set-up.....3

4 Deploying WAR file to Tomcat Server.....3

5 Set-up Nginx as a Reverse-Proxy.....4

1 Introduction

This document tells you how to deploy the Turbo Consent web application. Turbo Consent is a Spring Boot application built using Maven. It is currently deployed on an Oracle Cloud Virtual Machine running Ubuntu 16.04 (accessible using public IP address 132.145.41.170). These instructions will assume you already have a working virtual machine running Ubuntu 16.04+.

These instructions will explain how to connect Turbo Consent to a MySQL database server, how to build and deploy the WAR file to a Tomcat server and how to set-up Nginx as a reverse-proxy to Tomcat.

2 Clone the Git Repository

Navigate to the directory where you would like the Turbo Consent git repository to be saved then obtain a copy of the repository using the command:

```
$ git clone https://github.com/JiangYeap/turbo-consent.git
```

There is an up to date WAR file named *ROOT.war* present in the *target* directory of the repository. However, if you would like to make any changes to the source code, you can run the development tests and build a new WAR file from within the git repository using the commands:

```
$ sudo apt-get update
$ sudo apt-get install maven
$ mvn clean install
```

You should then rename the new WAR file using the command:

```
$ sudo mv TurboConsent-1.0-SNAPSHOT.war ROOT.war
```

3 MySQL Database Set-up

MySQL is the database management system that the Turbo Consent web application uses to manage its data.

To install MySQL, use the commands:

```
$ sudo apt-get update
$ sudo apt-get install mysql-server
```

Run MySQL using the command:

```
$ sudo mysql --password
```

Once you have selected a password, MySQL should run. Configure the Turbo Consent database using the following commands:

```
mysql> create database consentDB;
mysql> create user 'tcuser'@'localhost' identified by 'tcpass';
mysql> grant all on consentDB.* to 'tcuser'@'localhost';
mysql> quit
```

Once the Turbo Consent web application has been successfully deployed, the tables within consentDB will automatically be generated and filled with sample data.

4 Deploying WAR file to Tomcat Server

Apache Tomcat is a web server and servlet container that is used to serve Java applications.

To install Tomcat 8, run the commands:

```
$ sudo apt-get install default-jdk
$ sudo apt-get install tomcat8
```

Navigate to the *webapps* directory of your Tomcat server using the command:

```
$ cd /var/lib/tomcat8/webapps/
```

Then delete the sample webapp that is currently stored in the directory using the command:

```
$ sudo rm -r ROOT/
```

Navigate to the *target* directory of the cloned Git repository and copy the Turbo Consent WAR file to the *webapps* directory of your Tomcat server using the command:

```
$ sudo cp ROOT.war /var/lib/tomcat8/webapps/
```

When a new WAR file is copied into the *webapps* directory, Tomcat automatically runs and deploys the Java application. Restart the Tomcat server using the command:

```
$ sudo service tomcat8 restart
```

Tomcat uses port *8080* to accept conventional requests. Allow traffic to that port using the command:

```
$ sudo ufw allow 8080
```

As long as your virtual machine allows requests from external clients on port *8080*, the Turbo Consent web application should now be accessible by going to your domain or public IP address followed by *:8080* in a web browser:

```
http://server_domain_or_IP:8080
```

5 Set-up Nginx as a Reverse-Proxy

Nginx is open source software for web serving, reverse proxying, caching, load balancing, media streaming and more. It is deployed in front of our Tomcat server to handle requests from clients, separating the Turbo Consent web application from the task of web serving. Nginx will serve static files such as images, CSS and Javascript much faster than Tomcat.

To install Nginx, use the command:

```
$ sudo apt-get install nginx
```

Nginx uses port *80* to accept conventional requests. Allow traffic to that port using the command:

```
$ sudo ufw allow 'Nginx HTTP'
```

Now we will make Tomcat reverse to Nginx, so the user can access Turbo Consent on the Tomcat server via Nginx on port *80*. To do this, delete the *default* Nginx configuration file and create a new one called *tomcat.conf* using the command:

```
$ sudo rm default
$ sudo nano /etc/nginx/sites-available/tomcat.conf
```

Copy the following content into the file:

```
upstream tomcat {
    server 127.0.0.1:8080 weight=100 max_fails=5 fail_timeout=5;
}

server {
    listen 80 default_server;
    listen [::]:80 default_server;
    server_name server_domain_or_IP;
    root /var/lib/tomcat8/webapps/ROOT/;

    location / {
        proxy_set_header X-Forwarded-Host $host;
        proxy_set_header X-Forwarded-Server $host;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_pass http://tomcat/;
    }
}
```

Save the file then navigate to */etc/nginx/sites-enabled* and create a symbolic link to the *default* configuration file:

```
$ cd /etc/nginx/sites-enabled
$ ln -s /etc/nginx/sites-available/tomcat.conf
```

Test the Nginx server using the command:

```
$ sudo nginx -t
```

If no errors are reported, restart Nginx to implement your changes:

```
$ sudo systemctl restart nginx
```

As long as your virtual machine allows requests from external clients on port *80*, the Turbo Consent web application should now be accessible by going to your domain or public IP address in a web browser:

```
http://server_domain_or_IP
```

The Turbo Consent web application should now be successfully deployed on a Tomcat server with a reverse proxy and load balancing, providing high-performance and stability even with high user-traffic.