Servoy Ubuntu Deployment Guide

Ubuntu version 14+.x , Tomcat 8.x, Servoy 8+.x, Ng-client

Summary

1. Install Tomcat (Open Source Web Server and servlet container)
2. Install PostgresSQL
3. Create directory for Servoy Application Server
4. Create War from Servoy
5. Upload War
6. Final Notes

Install Tomcat

1. Enter server terminal as root user, and execute the following command to install tomcat 8 and other required packages.

```
sudo apt-get install ant git tomcat8 default-jdk
```

2. Modify the /etc/default/tomcat8 file and replace JAVA_OPTS with below. This adds the servoy application server folder and sets up the default amount of RAM allocated to Tomcat. Below are settings for systems with 512mb ram. Change the values as necessary for your setup.

```
sudo nano /etc/default/tomcat8
```

```
JAVA_OPTS="-DSERVOY_USER_HOME=/home/servoy
-Djava.security.egd=file:/dev/./urandom -Djava.awt.headless=true -Xms512m -Xmx512m
-XX:+UseConcMarkSweepGC"
```

3. By default Tomcat runs on 8080 port. You can change this by modifying the server.xml configuration file.

```
nano /etc/default/tomcat8/server.xml
```

```
<Connector port="8080" protocol="HTTP/1.1"
           connectionTimeout="20000"
           URIEncoding="UTF-8"
           redirectPort="8443" />
```

Install PostgreSQL

1. As root user, execute the following command to install postgres

```
sudo apt-get install postgresql postgresql-contrib
```

2. Change to postgres user

```
sudo -i -u postgres
```

3. Create default Servoy database (servoy repository)

```
createdb servoy_repository
```

4. Create additional databases as neccessary to support solution. eg.

```
createdb example_data
createdb mySchema
```

5. Create password for postgres user by executing the following commands and entering the postgres database interface.

```
psql
\password postgres
\q
```

6. Return back to root user

```
exit
```

Create Directory for Servoy Application Server

1. Create home directory for servoy

```
cd /home
mkdir servoy
```

2. Set permissions to allow read and write access

```
chmod 777 servoy
```

Create War file from Servoy

1. Open servoy and select solution to export
2. Choose export as ‘WAR export’.
3. Mark ‘include active solution and modules’ and ‘Allow data model changes’ - this will update the database schema with the changes required for the solution to run properly.
4. Setup a default user for the administration pages.
5. Configure databases and required plugins.

Upload the war

1. Stop the tomcat service if it’s running
`sudo service tomcat8 stop`

2. Then use sftp or scp to upload the .war file created from servoy to the webapps directory.
`scp mySolution.war root@myserver.com:/var/lib/tomcat8/webapps`

3. Restart or start the tomcat 8 service
`sudo service tomcat8 start`

Final Notes

service commands
`sudo service tomcat8 start` - start the service
`sudo service tomcat8 stop` - stop the service
`sudo service tomcat8 restart` - restart the service

directories
/var/lib/tomcat8/logs - tomcat logs
/var/lib/tomcat8/webapps - main web apps folder - can hot-deploy war files here
/home/servoy/.servoy/server/*mysolution/ - servoy logs and properties file

Starting web app
To access the application use the browser and go to admin page or run the solution.

http:\\myserver.com:8080*mysolution\servoy-admin
http:\\myserver.com:8080*mysolution\servoy-ngclient

Debugging Tomcat & Servoy

```
sudo tail -f /var/log/tomcat7/catalina.out
sudo tail -f /home/servoy/.servoy/server/*mysolution/servoy_logs.txt
```

http:\\myserver.com:8080*mysolution\servoy-admin
The admin page also contains useful logs and performance analysis for queries.