Deploy text-track-service to a new server

1. Ssh into your server ip and create a new user for the server with

*adduser username(texttrack).* Enter a password, rest of the fields can be left blank.

1. Now we need to add this user to the sudo users list so they can run root commands when needed by *usermod -aG sudo username(texttrack).*
2. Add your public key to the new user so you can ssh as texttrack user. Navigate to /home/texttrack. Create a .ssh folder via *mkdir .ssh.* Next make a file authorized\_keys via *cd .ssh and then touch authorized\_keys.* Now open this file via *sudo nano authorized\_keys. Add your public key here. Cmd/ctrl x to exit type y to save and hit enter.*
3. Switch to the new account via *su username(texttrack)* and test if you have sudo access via *sudo ls -la /root.* (will only work if you have sudo access).
4. Now cd enter and install bigbluebutton via the following command. You will need to do this as root so type exit and enter before the command.

*wget -qO- https://ubuntu.bigbluebutton.org/bbb-install.sh | bash -s -- -v xenial-220-beta -s bbb.example.com -e* [*info@example.com*](mailto:info@example.com)

Make sure to replace bbb.example.com with your domain name and [info@example.com](mailto:info@example.com) with your email. Also make sure to switch back to texttrack user after installing.

1. You can visit your domain after this at *bbb.example.com/demo/demo10.jsp* and start a meeting to see if it works.
2. Now to install the text-track-service rails app navigate to cd /usr/local and git clone from <https://github.com/bigbluebutton/text-track-service>.

sudo git clone <https://github.com/bigbluebutton/text-track-service>

1. Cd text-track-service. Make sure you have redis installed.
2. Now to install factory run the following two commands

sudo wget https://github.com/contribsys/faktory/releases/download/v1.0.1-1/faktory\_1.0.1-1\_amd64.deb

sudo dpkg -i faktory\_1.0.1-1\_amd64.deb

1. Now to configure the start scripts first cd development. Now

sudo ./setup.sh

This will configure the start scripts. You can move back by cd ..

1. Now we have to setup the rails app.

sudo apt install build-essential

sudo apt install autoconf bison build-essential libssl-dev libyaml-dev libreadline6-dev zlib1g-dev libncurses5-dev libffi-dev libgdbm5 libgdbm-dev

# Needed for rake db:setup

sudo apt install nodejs

sudo apt install npm

# setup rbenv

git clone https://github.com/rbenv/rbenv.git ~/.rbenv

echo 'export PATH="$HOME/.rbenv/bin:$PATH"' >> ~/.bashrc

echo 'eval "$(rbenv init -)"' >> ~/.bashrc

source ~/.bashrc

rbenv

git clone https://github.com/rbenv/ruby-build.git ~/.rbenv/plugins/ruby-build

rbenv install 2.6.1

rbenv local 2.6.1

ruby -v

gem install bundler

gem env home

gem install rails -v 5.2.3

rbenv rehash

apt-get install libsqlite3-dev

sudo chown -R texttrack.texttrack ~/.rbenv

sudo chown -R texttrack:texttrack /usr/local/text-track-service

bundle install

rails db:setup

# Start rails and listen on all interfaces

rails s -b 0.0.0.0

#To test

curl <http://localhost:3000> from a new terminal.

1. Now that your rails app is running open Terminal 2

# Copy example-credentials.yaml to credentials.yaml

cp example-credentials.yaml credentials.yaml

# Edit credentials.yaml to setup your credentials for the providers.

# Start the service

./development/start-service.sh

1. Open Terminal 3

# Start the worker

./developement/start-worker.sh

1. Open terminal 4 and navigate to /usr/local/bigbluebutton/core/scripts/post\_publish

Gem install rest-client

open post\_publish.rb and add the following code:



1. With all the 3 terminals running record a meeting and it should give you captions by the default ibm service. To change default open the settings.yaml file and edit the default in there.