

Installation

Audience: Client IT, Developers

Production

Installation on Ubuntu 16.04 LTS

1. Create the directory /usr/local/apps/

```
sudo mkdir /usr/local/apps
cd /usr/local/apps
sudo chmod 775 /usr/local/apps
sudo chgrp adm /usr/local/apps
```

- NOTE: The above assumes that your user has sudo privileges and is part of the `adm` group

2. Install Dependencies

```
sudo apt-get update
sudo apt-get install git python3 python3-virtualenv virtualenv gcc make
uwsgi uwsgi-plugin-python3 gdal-bin python-gdal python3-gdal python3-dev
build-essential -y
```

3. Clone repository into /usr/local/apps/

```
git clone https://github.com/Ecotrust/TEKDB.git
```

4. Create and activate a Python3 Virtual Environment

```
cd /usr/local/apps/TEKDB/
virtualenv env --python=python3
source /usr/local/apps/TEKDB/env/bin/activate
```

5. Install Python dependencies

```
pip install -r /usr/local/apps/TEKDB/TEKDB/requirements.txt
```

6. [Install Postgres + PostGIS](#)

- You will need to replace USERNAME in the steps below with a unique database user name
- The database name 'tekdb' below is assumed. Changing this will also require changes in the program settings.

```
sudo add-apt-repository "deb http://apt.postgresql.org/pub/repos/apt/  
xenial-pgdg main"
```

```
sudo add-apt-repository ppa:ubuntugis/ubuntugis-unstable
```

Press Enter to accept

```
wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo  
apt-key add -  
sudo apt update  
sudo apt install postgresql-9.6 postgresql-contrib-9.6 postgis  
postgresql-9.6-postgis-2.3 -y
```

Replace USERNAME with an appropriate username for your database user below.

```
sudo -u postgres createuser -s -P USERNAME  
sudo -u postgres createdb -O USERNAME tekdb  
sudo -u postgres psql -c "CREATE EXTENSION postgis; CREATE EXTENSION
```

```
postgis_topology;" tekdb
```

7. Enable External Access to Postgres Enable authentication from remote access

```
sudo vim /etc/postgresql/9.6/main/pg_hba.conf
```

Find the section of uncommented lines near the bottom. Edit the first of those lines to use 'trust' (instead of 'peer') authentication for user postgres on localhost:

```
local    all             postgres                                trust
```

Add the following line to the Authentication section at the bottom:

```
host     tekdb           USERNAME          0.0.0.0/0          md5
```

Enable remote access to PostgreSQL server:

```
sudo vim /etc/postgresql/9.6/main/postgresql.conf
```

Uncomment the 'listen_addresses' line and change it to read as such:

```
listen_addresses = '*'
```

Restart PostgreSQL Server:

```
sudo service postgresql restart
```

8. Install Proj.4

```
cd /tmp
wget http://download.osgeo.org/proj/proj-4.9.1.tar.gz
wget http://download.osgeo.org/proj/proj-datumgrid-1.5.tar.gz
tar xzf /tmp/proj-4.9.1.tar.gz
cd /tmp/proj-4.9.1/nad
tar xzf /tmp/proj-datumgrid-1.5.tar.gz
cd /tmp/proj-4.9.1
./configure
make
sudo make install
```

9. Update Local Settings

```
cp /usr/local/apps/TEKDB/TEKDB/TEKDB/local_settings.py.template
/usr/local/apps/TEKDB/TEKDB/TEKDB/local_settings.py
vim /usr/local/apps/TEKDB/TEKDB/TEKDB/local_settings.py
```

- You may use whatever text editor you prefer: emacs, nano, vim, etc...
- Set the following:
 - **ALLOWED_HOSTS**: add your domain name to the list, i.e.:

```
[
    'localhost',
    'your.site.com'
]
```

- default_lon: the longitude of the center of your study area in EPSG:3857 coordinates (get a GIS tech to help you with this)
- default_lat: the latitude of the center of your study area in EPSG:3857 coordinates
- default_zoom: 'Guess and check' - find the zoom level that shows your whole study area in the app's map view window. Here is an example of what these numbers look like:
 - 0: the whole world
 - 3: The U.S.A. or a small continent
 - 6: A medium-large U.S. state
 - 10: A county
 - 12: A forest
- SECRET_KEY: make something long and complex - you will not need to remember or enter it, just be sure not to share it.
- DATABASES:

```
{
  'default': {
    'ENGINE': 'django.contrib.gis.db.backends.postgis',
    'NAME': 'tekdb',
    'USER': 'USERNAME',
    'PASSWORD': 'PASSWORD',
    'HOST': 'localhost',
    'PORT': 5432
  }
}
```

10. Run migration

```
python /usr/local/apps/TEKDB/TEKDB/manage.py migrate
```

11. Run static file collection

```
python /usr/local/apps/TEKDB/TEKDB/manage.py collectstatic
```

12. Create Superuser (or import legacy data)

- If building a new database without data to load into it

```
python /usr/local/apps/TEKDB/TEKDB/manage.py createsuperuser
```

- If migrating old MTKEDB data in, [follow steps in this document].(<https://github.com/ECOTRUST/TEKDB/wiki/Migrating-From-Access-to-Postgres>)

13. Test your installation with Django's Dev Server

- If you need to debug and test here:
 - Make sure network traffic is allowed on port 8000
 - `run python /usr/local/apps/TEKDB/TEKDB/manage.py runserver 0.0.0.0:8000`
 - Know the IP address (or URL if you have DNS set up) of your server
 - punch your site's address at port 8000 into a browser, i.e.:
`your.domain.com:8000`

14. Install and configure NGINX

```
sudo apt-get install nginx
sudo cp /usr/local/apps/TEKDB/deployment/tekdb_nginx.conf
/etc/nginx/sites-available/tekdb
sudo rm /etc/nginx/sites-enabled/default
sudo ln -s /etc/nginx/sites-available/tekdb /etc/nginx/sites-enabled/tekdb
sudo cp /usr/local/apps/TEKDB/deployment/uwsgi_params /etc/nginx/
```

15. Configure UWSGI and boot proceses:

```
sudo cp /usr/local/apps/TEKDB/deployment/emperor.ini /etc/uwsgi/
sudo ln -s /usr/local/apps/TEKDB/deployment/uwsgi.service
/etc/systemd/system/
sudo ln -s /usr/local/apps/TEKDB/deployment/tekdb.ini
/etc/uwsgi/apps-enabled/
sudo service uwsgi start
sudo service uwsgi restart
sudo cp /usr/local/apps/TEKDB/deployment/rc.local /etc/rc.local
```

16. Set Media Folder Permissions:

```
sudo groupadd mediausers
sudo adduser www-data mediausers
sudo chgrp -R mediausers /usr/local/apps/TEKDB/TEKDB/media
sudo chmod -R 770 /usr/local/apps/TEKDB/TEKDB/media
```

17.Reboot

```
sudo reboot 0
```

18.Test Server and Networking/Ports

Debugging

- startup scripts logs: /var/log/rc.local/log
- nginx error log: /var/log/nginx/tekdb.error.log
- nginx access log: /var/log/nginx/tekdb.access.log

Development:

1. Create a development directory
 - i. this doc will assume Unix formatted addresses:
 - a. if running Windows, you will use \ instead of /
 - ii. this doc assumes the location /usr/local/apps/
 - a. Windows would looks something like C:\apps\

2. [Install Git](#)

3. Clone repository into /usr/local/apps/

```
cd /usr/local/apps/
```

```
git clone https://github.com/ECOTRUST/TEKDB.git
```

4. [Install Vagrant](#)

5. Deploy base box with Vagrant

```
cd /usr/local/apps/TEKDB/  
vagrant up
```

- i. Wait while your new VM installs all of the required dependencies
6. Log in to your new VM and run updates

```
vagrant ssh  
sudo apt-get update  
sudo apt-get upgrade
```

7. Update Local Settings

- i. File (same file, just accessed from either the host or the vm):
 - a. On Host server:
/usr/local/apps/TEKDB/TEKDB/TEKDB/local_settings.py
 - b. On Vagrant VM: /usr/local/apps/TEKDB/TEKDB/local_settings.py
- ii. Things to change:
 - a. SECRET_KEY: change to a long string of gibberish - you never need to type this in, so the longer and more complex the better
 - b. DATABASE_GEOGRAPHY: basic geo settings so the maps know where to focus by default in the admin
 - a. For example:

```
DATABASE_GEOGRAPHY = {  
    'default_lon': -124.2,  
    'default_lat': 41.9,  
    'default_zoom': 10,  
    'map_template': 'gis/admin/ol2osm.html',  
}
```

- b. NOTES:
 - a. LAT/LON: the lon and lat presented here are Coordinates (EPSG:4326) for the Northern CA coast. At the time of this writing, the tool only supports coordinates in Web Mercator (EPSG:3857), so the longitude would actually be -13825880.7546 and the latitude is 5146011.678566459.

- b. ZOOM: The higher the number, the more closely zoomed in the map will be by default.
- c. TEMPLATE: The location of the html template for displaying the map in the admin tool for geometry fields.

8. Activate your python virtual environment

```
source /usr/local/apps/TEKDB/env/bin/activate
```

9. Run migration

```
python /usr/local/apps/TEKDB/manage.py migrate
```

10. Run static file collection

```
python /usr/local/apps/TEKDB/manage.py collectstatic
```

11. Load in the dummy data

```
python /usr/local/apps/TEKDB/manage.py loaddata  
/usr/local/apps/TEKDB/TEKDB/fixtures/all_dummy_data.json
```

12. Test your installation with Django's Dev Server

```
python /usr/local/apps/TEKDB/manage.py runserver 0.0.0.0:8000
```

- i. Open <http://localhost:8111>