Installation

The MANRS benchmarking tool was developed on Ubuntu Linux but should work on most Linux and BSD systems and possibly on other systems which meet the following requirements. When examples/commands are used they are for the Ubuntu Linux operating system.

Requirements

System Packages

The following system packages are required and need to be installed:

- postgresql
- Python 3

Python 3

Although not necessary it is strongly advised to use `virtualenv` in order to install the Python dependencies in a Python virtual environment. If you want to skip the Python virtual environment go straight to step 4.

1. Install virtualenv:

   ```
   pip install virtualenv
   ```

2. Create the virtual environment with the python3 interpreter:

   ```
   virtualenv -p <path/to/the/python3/interpreter> <name_of_the_venv>
   ```

3. Activate the virtual environment:

   ```
   source name_of_the_venv/bin/activate
   ```

4. The required packages can be found in requirements.txt and can be installed by:

   ```
   pip install -r requirements.txt
   ```

If you selected to setup the virtual environment the following steps consider that the environment is activated whenever a Python execution takes place.

Setup

1. Create the database user (Note the password):

   ```
   sudo -u postgres createuser <db_user> -P
   ```

2. Create the database:

   ```
   sudo -u postgres createdb -O <db_user> <db_name>
   ```

3. Fill the appropriate values in the `config.py` file under the `POSTGRESQL_` options.

4. Create the schema in the database:

   ```
   python create_db.py
   ```

5. Setup a cronjob for gathering the CIDR data daily:

   a. `crontab -e` as a user that has read/write access to the project's directory;

   b. We need a cronjob that will run at least twice per day to make sure that we pick up changes from the CIDR website. Running the python script more than once per day is not destructive behavior because if we have already gathered the data for that day the script will just exit. A crontab entry could then be something like the following:

   ```
   22 */3 * * *  cd <project_directory>/cidr; <path_to_the_venv>/bin/python get_daily_
   ```

   c. Logging will be available in the `<project_directory>/cidr` directory.

6. Setup a cronjob for running the benchmark tool that generates the reports:

   a. `crontab -e` as a user that has read/write access to the project's directory;

b. We need the tool to run **once** per month. *Running the tool more than once per month will generate additional reports and mess with the data.* Running the tool with only the `-t auto` argument will gather and generate the report for the *previous month.* A crontab entry could then be something like the following:

```
22 15 1 * *  cd <project_directory>; <path_to_the_venv>/bin/python benchmark.py -t
```

c. Logging will be available in `<project_directory>/benchmark.log`.

7. Setup apache and enable the module `mod_wsgi`.

8. Configure apache to include something like the following in your <VirtualHost> directive:

```
WSGIDaemonProcess <name_of_the_process> python-home=<path_to_the_venv> python-path=<pat
WSGIProcessGroup <name_of_the_process>
WSGIScriptAlias / <path_to_project>/api.py process-group=<name_of_the_process>

<Directory <path_to_project>>
    Order allow,deny
    Allow from all
</Directory>
```

More information on configuring `mod_wsgi` can be found here.