

Building the project

This directory contains some sample config files that can be used a reference for setting up the project. Feel free to use them and change the variables accordingly.

Setting up Apache2

- `sudo apt install apache2 -y` - Install apache2
- `sudo apt install libapache2-mod-auth-openidc` - Installs library for running openid
- Enable the following mods: `ssl`, `auth_openidc`, `wsgi`, `rewrite`
- `sudo nano /etc/apache2/auth_openidc.conf` and copy the following to the file

```
OIDCProviderMetadataURL https://openidc.dcs.aber.ac.uk/auth/realms/MMP-IMPACS/.well-known/openid-configuration
OIDCClientID MMP-IMPACS
OIDCRedirectURI /oauth2callback #Replace with Link to website e.g. "discord.dcs.aber.ac.uk/oauth2callback"
OIDCCryptoPassphrase some-random-string-for-encrypting-cookies
OIDCScope "openid basic"
OIDCRemoteUserClaim preferred_username
OIDCSessionInactivityTimeout 86400
```

- `sudo cp ~/aberslink/config/aberslink.conf /etc/apache2/sites-available/` - copy the config file to the directory

6. Edit the file to match the file structure, below is an example of what to change the settings to:

```
<VirtualHost *:80>
    ServerName discord.dcs.aber.ac.uk
    ServerAlias discord.dcs.aber.ac.uk
    ServerAdmin cs-support@aber.ac.uk
    DocumentRoot /home/joel/aberslink/src/

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

- `sudo a2ensite aberslink` - Enable the website
- Go to `sudo nano /etc/apache2/apache2.conf` and add the following:

```
<Directory /home/joa38/aberslink/src/>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>
```

Setting up SSL

- Install the certbot package: `sudo apt intall certbot`
- Run this command to create the SSL certificates: `sudo certbot certonly --webroot -w ~/aberslink -d discord.dcs.aber.ac.uk`
- `sudo a2enmod rewrite` - Enable the rewrite mod for website redirect
- `sudo a2enmod ssl` - Enable the ssl mod for SSL to work on the website
- `source /etc/apache2/envvars` - fixes errors in apache2
- Open the conf file again (e.g. `aberslink.conf`) and change it match the domain name:

```
<VirtualHost *:443>
    ServerAdmin cs-support@aber.ac.uk
    serverName discord.dcs.aber.ac.uk
    ServerAlias discord.dcs.aber.ac.uk
    DocumentRoot /home/joa38/aberslink/src/AberLinkAuthentication

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    Alias /static /home/joa38/aberslink/src/AberLinkAuthentication/static
    <Directory /home/joa38/aberslink/src/AberLinkAuthent/static>
        Require all granted
    </Directory>

    <Directory /home/joa38/aberslink/src/AberLinkAuthentication/AberLinkAuthentication>
        <Files wsgi.py>
            Require all granted
        </Files>
    </Directory>

    <Directory /home/joa38/aberslink/src/AberLinkAuthentication>
        Options Indexes FollowSymLinks
        AllowOverride None
        Require all granted
        Allow from all
    </Directory>

    <Location />
        AuthType openid-connect
        Require valid-user
    </Location>

    WSGIScriptAlias / /home/joa38/aberslink/src/AberLinkAuthentication/AberLinkAuthentication/wsgi.py
    WSGIDaemonProcess django_app python-path=/home/joa38/aberslink/src/AberLinkAuthentication python-home=/home/joa38/ab
    WSGIProcessGroup django_app

    SSLEngine on
    SSLCertificateFile /etc/letsencrypt/live/discord.dcs.aber.ac.uk/fullchain.pem
    SSLCertificateKeyFile /etc/letsencrypt/live/discord.dcs.aber.ac.uk/privkey.pem
    SSLProtocol all -SSLv2 -SSLv3 -TLSv1 -TLSv1.1
</VirtualHost>
```

- `sudo apachectl restart` - Restart Apache2 to enable the new mods

installing the dependencies for the bot

- `sudo apt install libpq-dev python-dev` - needed to install `psycpg2-binary`
- Navigate to the folder `aberslink/src/AberLinkDiscord` and type `pipenv install`
- Create a discord bot token by visitng <https://discord.com/developers/applications> and creating a new bot called AberLink along with the supplied photo `/img/AberLink_logo_cropped.png`. Then head on over to the Bot panel and create a new bot with the same information as above. Finally copy the token underneath the bots name for the next section.
- While inside the folder `/aberslink/src/AberLinkDiscord` create a new file using `nano .env` and add the following (or copy the example provided in this folder):

```
DISCORD_TOKEN= # Discord token found on bot's page
DATABASE_NAME= # Postgres database name
USER= # Postgres user
PASSWORD= # Postgres password
HOST= # Postgres host
PORT=5432 # This is the default port
WEBSITE_URL= # The website for AberLinkAuthentication e.g. https://joa38-mmp.dcs.aber.ac.uk/
```

- Check that the .env file has been configured correctly by typing in the command `pipenv run python3 -m AberLink`.
- Invite the bot to one of your servers by creating an invite link in the tab OAuth2 and selecting `bot` in the `scopes` section and then in the `bot permissions` tick `Administrator`. Copy the link generated by it and paste it into the browser and invite the bot to your server.
- In the OAuth2 tab `Redirects` section add your website uri e.g. `https://discord.dcs.aber.ac.uk/oauth2/login/redirect`.
- On the same page scroll down to the `OAuth2 URL Generator` and select the web address you entered in the `Redirects` section.
- Scroll down again and in the `scopes` section select `identify` and copy the url. Then head on over to the file `/aberslink/src/AberLinkAuthentication/login/views.py` and find the `discord_oauth2` function and replace the redirect URL with your own.

Installing and running the webserver

- navigate to the folder `aberslink/src/AberLinkAuthentication` and type `virtualenv venv` to create a virtual enviornment folder for the data.
- `source venv/bin/activate` - activates the virtuanenv
- `pipenv install` installs dependencies from the project into the file
- Make sure that that the venv file path is correct inside of the `aberslink.conf` file.

Django setup

- `sudo cp config/config.json /etc/config.json` - copy the template file for the django config and fill out the details below (email `joa38@aber.ac.uk` for `SECRET_KEY`):

```
{
    "SECRET_KEY": "",
    "DATABASE_NAME": "",
    "USER": "",
    "PASSWORD": "",
    "HOST": "",
    "PORT": "",
    "DISCORD_CLIENT_SECRET": "",
    "DISCORD_TOKEN": "",
    "WEBSITE_URL": ""
}
```

Note: The `WEBSITE_URL` is the name of the website that is going to be used. e.g. `https://joa38-mmp.dcs.aber.ac.uk`

The `DISCORD_CLIENT_SECRET` and `DISCORD_TOKEN` can be found by visitng the page created earlier for the discord bot.

- Navigate to the `General Information` tab and copy the client secret which is located below the Description on the right hand side. Copy and save this variable to the `DISCORD_TOKEN`
- Naviage to the `Bot` tab and copy the name token located below the username and save it to the `DISCORD_CLIENT_SECRET`

After setting up the config file open the command shell and type the following commands to configure the database for Django:

- `python3 manage.py makemigrations`
- `python3 manage.py sqlmigrate`
- `python3 manage.py migrate`

Finally once everything has been ensured to be fully functioning open the file

`src\AberLinkAuthentication\AberLinkAuthentication\settings.py` and find the line `DEBUG = True` and set the variable to `False`