ASReview server for MegaMeta study

Description of the Virtual Machine

- Virtual Machine is provided by the Faculty of Science, University of Amsterdam.
- At the beginning it is 'large' machine with 4 CPUs, 8Gb and 10Gb SSD
- The person I was in contact with to set up the machine is Auke Folkerts.
- There are no root or super-user privileges, so much of a configuration was done through Auke. This included:
    - Setting up the host name `ivi-megameta.science.uva.nl`
    - Setting up Nginx to work with SSL and forward the requests coming to 443 (HTTPS) port to whatever process listening on 8080 (HTTP) port.
    - This seamless use of HTTPS is a good thing, since we are using basic HTTP-Auth mechanism, which does not have man-in-the-middle protection by itself, so HTTPS protects the password sent openly from being easily read by a third-party.
    - Setting up the service of the server for it to start with a system startup and being easily stopped and started and quired for the status.
- Machine can only be accessed from within UvA network, so UvA account and VPN is required.
- Log-in to the machine is done via SSH via

```
ssh vmelnik@ivi-megameta.science.uva.nl
```

- The home directory in which all the files are stored is `/home/vmelnik/`

Setup of the ASReview

ASReview was cloned into `~/asreview/` with the command

```
mkdir ~/asreview
cd ~/asreview
git clone -b http-auth https://github.com/valmelnikov/asreview.git
```

Python 3.8 virtual environment was created at `~/asreview/` and activated through

```
/opt/rh/rh-python38/root/usr/bin/python3.8 -m venv pyenv
source pyenv/bin/activate
```

ASReview dependencies were installed through

```
python setup.py install
```

Then the package was removed with

```
cd ~
python -m pip uninstall asreview
```

Gunicorn was installed

```
python -m pip install gunicorn
```

The React app was built from the same repository on a local machine (due to unavailability of NPM on VM)

```
# On a local machine. Basically the commands from
/path/to/asreview/asreview/webapp/compile_assets.sh
cd /path/to/asreview/asreview/webapp
npm install
npm run-script build

# Copying to VM

scp -r /path/to/asreview/asreview/webapp/build vmelnik@ivi-
megameta.science.uva.nl:/home/vmelnik/asreview/asreview/webapp/
```

In the easiest scenario the setup can be checked through running (while `~/asreview/pyenv` is activated)

```
cd ~/asreview
source asreview/bin/activate
python -m asreview lab --port 8080
```

which is what is written in `~/run_asreview.sh`

The unicorn-served production scenario is invoked as follows:

```
cd ~/asreview
source asreview/bin/activate
gunicorn -w 4 -b :8080 --timeout 120 'asreview.webapp.start_flask:create_app()'
```

Where `-w 4` stands for 4 worker processes matching the number of CPUs. These commands are the contents of `~/run_gunicorn.sh`. Similar script is used to define the gunicorn service

```
ExecStart=/home/vmelnik/asreview/pyenv/bin/gunicorn -w 4 -b :8080 --timeout 120
'asreview.webapp.start_flask:create_app()'
```

And then the service is operated with the following commands:

```
sudo /usr/sbin/service vmelnik-gunicorn stop
sudo /usr/sbin/service vmelnik-gunicorn start
sudo /usr/sbin/service vmelnik-gunicorn status
```

Important notes

- The setup of the projects only works with `asreview lab` command, not gunicorn server. Most probably it is because important declarations for file readers are done when invoking `asreview lab`, but not when directly asking Flask app from `start_flask.py`.
- Maximum file size setting of Nginx was increased to 500 Mb.

Setting up passwords

The branch `http-auth`, from which the repository was cloned, was implemented to make HTTP-Auth possible to add another level of protection and enable project-locking functionality.

The password for the user `user1` is set (for the new user) and changed (for the existing user) with the command:

```
cd ~/asreview
source pyenv/bin/activate
python -m asreview auth -u user1
```

Which will prompt to the entry of password. To delete the user the following command is used

```
python -m asreview auth -u user1 -r
```

Note that for the auth credentials being updated on the running server, it needs to be restarted.

The passwords are stored in the file at standard path `~/.asreview/auth.txt`

Accessing the ASReview

Once under UvA VPN or any other permitted network, the server can be accessed from any browser by navigating to `https://ivi-megameta.science.uva.nl` and entering the credentials, which should be sent to each intended user separately.

Project files and backups

The ASReview project files are stored at default location `~/.asreview/`.

There are two types of backups being made:

- Full system backup being done daily for the last 14 days. Rolling back to it will require asking responsible person (Auke) and will affect all the progress, which was made since the date of backup we are rolling back to.
- Per-project backups scheduled on `cron`
    - script run by cron is `~/backup.sh`
    - once a day at 21:00

      ```
      # Running
      crontab -e
      # Reads
      0 21 * * * /home/vmelnik/backup.sh
      ```

    - stored at `~/backup/incr/PROJECT_NAME/DATE_TIME`
    - made with `rsync` tool
    - incrementally, meaning that only files changed since the last backup are included in the new backup. This helps to save space.
    - It also means that the full backup is the first one, thus, **removing the older backups to save space is not a valid approach, which will render all the backups useless**.

      Note from 21/09/2021: It is actually a valid approach due to rsync using links to files rather than hard copies. When removing older folders, the files are not deleted because they are refered to from the newer folders. So, something like below should work fine (but better to check on one project first):

      ```
      cd ~/backup/incr/PROJECT_NAME
      ls -v . | head -n -14 | xargs rm -r
      ```

    - Space-wise each backup so far takes ~60Mb if there are changes, for 3 projects for 2 times a day it should amount to 360Mb, 7.5Gb for 20 days. But this calculation is only fair if project files do not grow in size with the review progress.
    - Thus, if there is a server error, one can check if the disk space is enough and ask Auke to remove all the backups from `~/backup/incr/` (if sure that they are not needed).
    - An example of restoring the state of the project is given in `~/example_restore_cmd.txt`. As is done there, it is better to first restore the files to some third location, check their consistency and then only `cp -rf ...` to the project folder at `~/.asreview/`