DReSA Technical Notes

Version 1.4.2

Contents

Instructions for installing and configuring the TeSS web application are available via the repository: [TeSS Repository]

A. Authentication via OpenID Connect

1. Install keybase & create an account

```
curl --remote-name
https://prerelease.keybase.io/keybase_amd64.deb; sudo apt install
-y ./keybase_amd64.deb; run_keybase
keybase login                # include account id and password
 keybase passphrase set      # set account passphrase
```

2. Register a new service with AAF. Provide the following details:
   - The service's *redirect URL* in the following format:
     `https://<url or ip address>/users/auth/oidc/callback`
   - A descriptive name for the service. e.g. '*National Training Registry and Calendar*'.
   - The organisation name, which must be an **AAF** subscriber, of the service, e.g., '*Pawsey Supercomputing Centre*'.
   - Indicate the service's purpose - development/testing/production-ready.
   - Your keybase account id to share the credentials securely (as defined above).

3. Upon notification of registration, copy the following parameters.

   `keybase chat read` # select message from AAF

   Copy the following parameters into `config/secrets.yml`:

```
external_api_keys:
  oidc:
   redirect_uri: <Redirect URI – as above>
     client_id: <Client ID>
     secret: <Secret>
     issuer: ''
      host: 'central.test.aaf.edu.au'
```

4. Restart Unicorn

```
sudo service unicorn_tess restart
```

OpenID Connect: Service information

- Organization: Pawsey Supercomputing Research Centre (*https://pawsey.org.au/*)
- Attribute Scopes: `openid, email, profile.`

Services:

- Federation: `Production`
  Redirect URL: `https://dresa.org.au/users/auth/oidc/callback`
  Service name: `Digital Research Skills Australasia`
  Service Purpose: `Production-ready`
  Landing Page: `https://dresa.org.au/`

- Federation: `Production`
  Redirect URL: `https://staging.dresa.org.au/users/auth/oidc/callback`
  Service name: `Digital Research Skills Australasia (Staging)`
  Service Purpose: `Staging / User acceptance testing`
  Landing Page: `https://staging.dresa.org.au/`

- Federation: `Test`
  Redirect URL: `https://test.dresa.org.au/users/auth/oidc/callback`
  Service name: `Digital Research Skills Australasia (Test)`
  Service Purpose: `Testing / Development`
  Landing Page: `https://test.dresa.org.au/`

B. Google Analytics & Maps

Analytics

1. Create a google analytics account at: `analytics.google.com`
   - Add Data Stream: Admin -> Data Streams -> Add stream (Web)
     - Add URL and Stream Name
     - Set properties measured, including:
       - Click: Link_url & links_domain
   - Use the 'Measurement Id' as `<google analytics code>` below.

Maps

2. Create a google developer profile at: `developers.google.com`
   - Set up a project: e.g. DReSA
   - Add Billing Details.
   - Use the API Key as `<google maps api key>` below.
3. Set the following properties in: `vim config/tess.yml`

```
gmaps:
  center:
    latitude: -33
    longitude: 150
  zoom:
    latitude: 3
    longitude: 10
```

Update TeSS

4. Edit environmental properties: `/etc/environment`
   - `PRODUCTION_GANAL_CODE=<google analytics code>`
   - `PRODUCTION_GMAPS_KEY=<google maps api key>`
5. Restart Unicorn: `sudo service unicorn_tess restart;`

Capture Outbound Links

Add the following parameters to an anchor/link:

- `href ="link_url"`
- `target="_blank"`
- `onclick="getOutboundLink('#{link_url}'); return true;"`

C. Scheduling Tasks

Install Cron service

- `sudo apt install cron`
- `sudo systemctl enable --now cron`

Default job timings

Job	Frequency	Time*
Sitemap	Day	3:30 pm
Subscriptions	Day	4 pm
Ingestions	Day	5 pm
Logrotate	Tuesday	3 pm
Backups	Monday	3 pm

* Time is as of the system time of the machine. Use the following command to check: `timedatectl`

Change job timings

- Copy examples: `cp config/schedule.example.yml config/schedule.yml`
- Edit entries in: `config/schedule.yml`

Add jobs to crontab

- Update crontab (production) during set up or after any change to job timings:

  `whenever --update-crontab --set db_user="$PRODUCTION_DB_USER"`

- Check crontab: `crontab -l`
- Create log file: `touch /home/ubuntu/TeSS/shared/log/cron.log`

For more information see: http://github.com/javan/whenever

Ensure scripts are executable

- `chmod 700 scripts/*.sh`
- `dos2unix scripts/*.sh`

New Jobs

- New tasks can be defined in the file: `lib/tasks/tess.rake`
- New jobs can be scheduled in the files: `config/schedule.rb`

Troubleshooting

- Cron sets its own path for commands, which can result in 'command not found' error. Sometimes you may need to specify the full path to the command.

D. Email Subscriptions

Enable the subscription feature

- `vim config/tess.yml`

  ```
  dresa.feature.subscription: true
  mailer.delivery_method: smtp
  ```

Set up scheduled tasks

- Push schedule.rb jobs to crontab:

  ```
  whenever --update-crontab --set environment='<environment>'
  ```

- Check crontab update: `crontab -l`
- Create log file (if not exists): `touch /home/ubuntu/TeSS/shared/log/cron.log`

Set up email service (using Google)

- Create an app password for google account: Sign in with App Passwords
- Add environment properties: `sudo vim /etc/environment`

  ```
  PRODUCTION_GMAIL_USERNAME=<google username>
  PRODUCTION_GMAIL_PASSWORD=<google app password>
  ```

- Add SMTP details: `vim config/secrets.yml`

  ```
  smtp:
    :user_name:    <%= ENV["PRODUCTION_GMAIL_USERNAME"] %>
    :password:     <%= ENV["PRODUCTION_GMAIL_PASSWORD"] %>
    :domain:               gmail.com
    :address:              smtp.gmail.comsu
    :port:                 587
    :authentication:       plain
    :enable_starttls_auto: true
  ```

- Restart Unicorn: `sudo service unicorn_tess restart;`

Previewing emails

Once generated, subscription emails can be viewed at the following address:

- `/rails/mailers/subscription_mailer`
  - `/last_event_digest`
  - `/last_material_digest`

E. Email Service

The following steps are required to set up an outbound email service. This will change the source of emails to the appropriate domain with valid certificates, which may help emails get through spam filters.

Install Postfix

- Install mailutils: `sudo apt install mailutils`
  - select option: `Internet Site`
  - enter domain, e.g.: `test.dresa.org.au`

- Verify the hostname in file: /etc/mailname

- Modify the configuration file: `/etc/postfix/main.cf`
  - `myhostname = <host name>`
  - `mydomain = <domain name>`
  - `inet_interfaces = loopback-only`
  - `inet_protocols = ipv4`
  - `mydestination = localhost.$mydomain, localhost, $myhostname`
  - `masquerade_domains = $mydomain`

- Restart Postfix: `sudo systemctl restart postfix`

Testing the SMTP Server

- `echo "Test content!" | mail –s 'Test Subject Line' <email address>`

Enable Encryption

- Modify the configuration file: `/etc/postfix/main.cf`

  `smtpd_tls_cert_file=/etc/ssl/certs/<bundle file>`

  `smtpd_tls_key_file=/etc/ssl/private/<private key>`

  `smtpd_tls_security_level=may`

- Restart Postfix: `sudo systemctl restart postfix`

Set Add sendmail details:

- Edit the configuration file: `config/tess.yml`

  `mailer.delivery_method: sendmail`

- Restart Unicorn: `sudo service unicorn_tess restart;`

F. Automated Ingestion

Create Configuration File

Copy ‘config/ingestion.example.yml’ to ‘config/ingestion.yml’

Edit 'config/ingestion.yml'

Add the following parameters, with examples:

```
name: 'production'           #
logfile: 'log/ingestion.log' # location of the log file
loglevel: 0                  # level of logging information
username: 'scraper'          # name of user with role 'scraper_user'
sources:                     # see below
```

Log Levels

0. All messages.
1. Task messages.
2. + Source validation messages.
3. + Ingestor summary messages.
4. + Resource summary messages.
5. + Resource detail error messages.

Add Sources

For each source add the following parameters:

```
- provider: ''      # the content provider's title
  url: ''           # the accessible url of the ingestion source
  method: ''        # one of 'csv', 'ical', or 'rest'
  resource_type: '' # one of 'event', or 'material'
```

Run the Task

```
rake tess:automated_ingestion
```

G. Maintenance Tasks

Update Root Certificate

Add certificate to root certificates

1. `sudo openssl x509 -outform der -in CERTIFICATE.pem -out CERTIFICATE.crt`
2. `sudo cp CERTIFICATE.crt /usr/local/share/ca-certificates`

Update the root certificates on a server

3. `sudo update-ca-certificates`
4. `sudo service nginx restart`

Override 'httpclient' root certificates

5. Check gem locations with: `gem which rubygems`
6. `ln -sf /etc/ssl/certs/ca-certificates.crt <TeSS>/vendor/bundle/ruby/<version>/gems/httpclient-<version>/lib/httpclient/cacert.pem`

Bump Ruby Version

Update the ruby version when required by gem version updates.

7. `cd TeSS`
8. `sudo apt update -y`
9. `[sudo apt upgrade -y]`
10. `cat /etc/postgresql/12/main/postgresql.conf`
    `check ssl = off, if not reset and restart postgresql service`
11. `git stash; git pull origin master`
12. `rvm install `cat .ruby-version``
    - If installation fails try the following:
    - `rvm list known` # and select an appropriate version, such as '2.6'
    - `rvm install ruby-head-<version>`
      `--url https://github.com/github/ruby.git`
      `--branch version`
13. `rvm --default use `cat .ruby-version``
14. `rvm list`
15. `rvm delete <old-ruby-version>`
16. `rvm --create `cat .ruby-version`@`cat .ruby-gemset``
17. `gem install bundler`
18. `rm -rf .bundle`
19. `rm -rf Gemfile.lock`
20. `bundle install`
21. `chmod 700 ./update_production`

22. `sh ./update_production`

Note: if rails command not available, may need to reinstall rails.

23. `gem install rails`

Install Yarn (if required)

1. `curl -sL https://dl.yarnpkg.com/debian/pubkey.gpg | sudo apt-key add -`

2. `echo "deb https://dl.yarnpkg.com/debian/ stable main" | sudo tee /etc/apt/sources.list.d/yarn.list`

3. `sudo apt update && sudo apt install yarn`

4. `yarn --version`

Database backup

1. `chmod 700 /home/ubuntu/TeSS/scripts/*.sh`
2. `mkdir /home/ubuntu/TeSS/shared/backups`

The following scripts can be used to backup and restore the database:

- `sh scripts/pgsql_backup.sh`
- `sh scripts/pgsql_restore.sh`

Or, you can run the backup script as follows:

- `sh scripts/pgsql_backup.sh tess_user tess_development ./shared/backups --exclude-schema=audit`
-

The parameters are as follows:

1. user: e.g. *tess_user*
2. database: e.g. *tess_development*, *tess_production*
3. backup folder: e.g. ./shared/backups
4. addtional parameters: e.g. --excluded-schema=audit

Note: sql files are stored with timestamped names as follows:

- folder/database.YYYYMMDD-HHMMSS.[schema,data].sql
- eg. ~/TeSS/shared/backups/tess_development-20210524-085138.data.sql

And, you can run the restore script as follows:

- `sh scripts/pgsql_restore.sh tess_user tess_production ./shared/backups/tess_production.20210524-085138.schema.sql ./shared/backups/tess_production.20210524-085138.data.sql`

With the parameters:

1. user: e.g. *tess_user*
2. database: e.g. *tess_development*, *tess_production*
3. schema file: e.g. ./shared/backups/tess_development.20210524-085138.schema.sql
4. data file: e.g. ./shared/backups/tess_development.20210524-085138.data.sql

Note: these scripts have been adapted from the repository: [fabioboris/postgresql-backup-restore-scripts]

- made available under the MIT License (MIT)
- Copyright (c) 2013 Fabio Agostinho Boris <fabioboris@gmail.com>

Enable PostgreSQL access

If access problems persist with Postgres:

1. Edit file: `/etc/postgresql/12/main/pg_hba.conf`

   Set the following lines to 'trust':

   ```
   # IPv4 local connections:
   host    all     all     127.0.0.1/32    trust
   # IPv6 local connections:
   host    all     all     ::1/128         trust
   ```

2. Restart postgres service:

   ```
   sudo service postgresql restart
   ```

Log File Rotation

Log files and their rotations are configured in the following file: `config/logrotate.conf`

Database list tables

```
1. sudo -i -u postgres
2. psql
3. \l                    # list databases
4. \c tess_production      # connect to database
5. \dt                       # list tables
6. select count(*) from <table>;
7. psql -d tess_production -U <username>
```

Ports

- WEBrick: 3000
- Redis: 6379
- PostgreSQL: 5432
- Nginx: 80, 443 (ssl)
- Solr: 8983 (production), 8982 (development), 8981 (test)

Issues

1. Problem: Rails server not visible outside the machine via WEBrick.
   Solution: Start rails with option: `bundle exec rails server -b 0.0.0.0`
2. Problem: Mimemagic version not available.
   Solution: Refresh mimemagic sources: `bundle update mimemagic`
3. Problem: PostgreSQL createuser requires a role with login rights.
   Solution: Use appropriate commands before setting the password [2].
4. Problem: RVM not installed.
   Solution: Follow instructions [3]
5. Problem: Nginx server PID error.
   Solution: Run workaround [6].
6. Problem: Tuakiri certificate error:
   Solution: Update certificate in httpclient gem [12]

Social Media

Twitter Image Sizes

- The ideal image size for Twitter Cards is **800px by 418px** (1.91:1 ratio).
- For App Cards, you can go with **800px by 800px** (1:1 ratio).
- Twitter supports images that are **JPEG or PNG format**; no GIFs are allowed here.
- For best results, make sure your image is **no larger than 3 MB.**
- Twitter Images Size Guide for 2021 | Adobe Spark

Resource Filters

The resource fields that are available to Solr are defined in the model's searchable method:

- `app/models/<model>.searchable()`

The list of fields that can be displayed on the sidebar are defined in the class method:

- `app/models/<model>.self.facet_fields()`

The list of facet fields that are ignored for filtering are defined in the following:

- `config/initializers/hidden_filters.IGNORED_FILTERS`

Re-initialise and reindex Solr after updating these properties:

- `sudo service unicorn_tess restart;`
- `rake sunspot:solr:reindex RAILS_ENV=<environment>`

H. Links

Training eSupport System

1. TeSS: https://github.com/nrmay/TeSS#readme
2. PostgreSQL: https://www.digitalocean.com/community/tutorials/how-to-install-and-use-postgresql-on-ubuntu-18-04

Ruby on Rails

3. RVM package for Ubuntu: https://github.com/rvm/ubuntu_rvm
4. Rails ERD: https://voormedia.github.io/rails-erd/install.html
5. Rails via Unicorn and Nginx: How To Deploy a Rails App with Unicorn and Nginx on Ubuntu 14.04 | DigitalOcean
6. DataTimePicker: eonasdan-bootstrap-datetimepicker - npm (npmjs.com)

Nginx, SSL and Certificates

7. Nginx: https://www.digitalocean.com/community/tutorials/how-to-install-nginx-on-ubuntu-18-04
8. Nginx Service Bug: https://stackoverflow.com/questions/42078674/nginx-service-failed-to-read-pid-from-file-run-nginx-pid-invalid-argument
9. Create Self-Signed SSL Cert for Nginx: https://www.digitalocean.com/community/tutorials/how-to-create-a-self-signed-ssl-certificate-for-nginx-in-ubuntu-18-04
10. Nginx – install certificates: How to install an SSL certificate on a NGINX server – HelpDesk | SSLs.com
11. Update certificates: https://support.kerioconnect.gfi.com/hc/en-us/articles/360015200119-Adding-Trusted-Root-Certificates-to-the-Server
12. Update httpclient certificate: https://github.com/nahi/httpclient/issues/445
13. How to troubleshoot RubyGems and Bundler TLS/SSL Issues: https://bundler.io/guides/rubygems_tls_ssl_troubleshooting_guide.html

Authentication

14. AAF OpenID: OpenID Connect (OIDC) : AAF Support
15. Omniauth OpenID Connect: https://github.com/m0n9oose/omniauth_openid_connect
16. AAF Validator: https://validator.test.aaf.edu.au/snapshots/latest
17. Tuakiri: Tuakiri OpenID Connect Bridge - Tuakiri - Tuakiri Confluence
18. Overview of JWT: https://redthunder.blog/2017/06/08/jwts-jwks-kids-x5ts-oh-my/

Ubuntu Services

19. Cron Jobs: How to Automate Regular Tasks with Cron on Ubuntu 20.04 (serverspace.io)
20. Logrotate setup: https://www.vultr.com/docs/using-logrotate-to-manage-log-files

Google Services

21. Google analytics set up on Rails 5: https://michaelsoolee.com/google-analytics-rails-5/
22. Google Calendar for Developers: https://developers.google.com/calendar/api/v3/reference/events
23. Google Calendar render parameters: https://github.com/InteractionDesignFoundation/add-event-to-calendar-docs/blob/master/services/google.md
24. Capture outbound links: https://support.google.com/analytics/answer/7478520?hl=en

Other

25. Meta Tags Viewer: OpenGraph - Preview Social Media Share and Generate Metatags
26. Twitter card validator: https://cards-dev.twitter.com/validator
27. PHP-Markdown Style: https://daringfireball.net/projects/markdown/