

Dariah contribtool Installation

- if something goes wrong after an unusual update

Set selinux momentarily in permissive mode.

Navigate to inkind, verify that everything works.

The switch to enforced mode again.

It happens when the python interpreter needs to recompile certain scripts.

It needs to generate an `__init__.pyc` here and there.

Packages needed

```
git vim python-pip python-lxml python-dev python-software-properties
curl make g++ zlib1g-dev binutils autoconf automake libtool
libmysqlclient-dev
```

```
pip install markdown
```

File System and commands

Make a group *dariah*, add user *dirkr* to that group.

Make directories `/opt-webapps` and `/opt/dariah`. These must belong to the group *dariah*. Sticky bit.

Make the following files in `/opt/dariah`:

- `pwd.cfg`
- `host.cfg`

These are for the database configuration. See later on.

User xxx may run the following commands on this host:

(root) NOPASSWD:

```
sudo /usr/bin/systemctl start httpd.service
sudo /usr/bin/systemctl restart httpd.service
sudo /usr/bin/systemctl stop httpd.service
sudo /usr/bin/systemctl status httpd.service
sudo /usr/sbin/apachectl configtest
sudo /sbin/reboot
```

MariaDB CONFIG

Make sure mariadb works with utf8.

Set the default-character-set to utf8

and `max_allowed_packet` to 33554432.

MariaDB USERS AND DATABASES

There will be the following databases:

dariah_web, which is the database with user data and

dariah_data, which contains the data that the dariah application gives access to.

There will be the following users:

dariah acting on behalf of the dariah web application.

It will have read/write access to the **dariah_web** and **dariah_data** databases.

dariah_admin acting as the database admin for the **dariah_web** and **dariah_data** databases.

This application will connect to the database server as **dariah**. Its password must be stored in /opt/dariah/pwd.cfg and its hostname in /opt/dariah/host.cfg. These files are expected to be single line files, with just the expected value in that line. The terminating newline will be ignored.

```
GRANT ALL PRIVILEGES ON `dariah_%`.* TO
'dariah_admin'@'applicationserver';
```

```
CREATE USER 'dariah'@'applicationserver' IDENTIFIED BY 'pwd';
```

```
GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, ALTER ON
dariah_web.* TO `dariah`@`%`;
GRANT SELECT, INSERT, UPDATE, DELETE ON dariah_data.* TO
`dariah`@`%`;
FLUSH PRIVILEGES;
```

WEB2PY

Create and store a new password for the web2py admin app (max 16 chars). We refer to this password as pwd-web2py-admin.

Here are instructions for setting up a production site.

Take care to have a proper certificate ready.

```
wget https://github.com/web2py/web2py/raw/master/scripts/setup-
web2py-fedora.sh
chmod +x setup-web2py-fedora.sh
```

If you are on Ubuntu, use setup-web2py-ubuntu.sh instead.

Edit the shell file. Skip the self-signed certificate. But do the selinux commands for the writable directories.

Make sure web2py ends up in /opt/web-apps/web2py

```
sudo ./setup-web2py-fedora.sh
```

Look at the lines specifying the certificate:

```
SSLCertificateFile /etc/apache2/ssl/inkind.domain.cer
SSLCertificateKeyFile /etc/apache2/ssl/inkind.domain.key
SSLCertificateChainFile /etc/apache2/ssl/CHAIN-
TERENA_SSL_CA_2.pem
```

and place these files in the indicated directories.

The script asks for admin password in the end. Take an arbitrary password for this. If you want to change that later, do the following.

```
cd /home/www-data/web2py
sudo -u www-data python -c "from gluon.widget import console;
console();"
sudo -u www-data python -c "from gluon.main import
save_password; save_password(raw_input('admin password: '),
443)"
```

End change password.

DARIAH

Here comes the actual **dariah** web application! We clone it from github.

For updates, we pull from github. This can be done on the command line, but also remotely, on the web2py administrative app.

We put the **dariah** app next to web2py, and link to it symbolically from the web2py/applications directory.

```
pushd /opt/web-apps
git clone https://github.com/etcbc/dariah
pushd web2py/applications
ln -s /opt/web-apps/dariah dariah
popd
popd
```

Check it out:

<https://dariah.domain> - the dariah contribution site

<https://dariah.domain/appadmin> - dariah maintenance

<https://dariah.domain/admin> - web2py admin app.

NOTES

Using the web2py admin app, it is possible to *compile* the application. It speeds up the application.

However, if you update views, you have to recompile, because the compilation step freezes the view inclusions. The recompile step is coded in the *update.sh* script.

APPENDIX

A: config file for web2py: routes.py in web2py root dir

```
routers = dict(
    BASE = dict(
        default_application='dariah',
        root_static=['robots.txt'],
    )
)
```

B: sample apache config file

```
#NameVirtualHost *:80
```

```
#NameVirtualHost *:443
```

```
#Include /etc/httpd/conf.d/shib.conf
```

```
<VirtualHost *:80>
```

```
    WSGIDaemonProcess web2py user=apache group=apache  
    processes=1 threads=1
```

```
    WSGIProcessGroup web2py
```

```
    WSGIScriptAlias / /opt/web-apps/web2py/wsgihandler.py
```

```
    WSGIPassAuthorization On
```

```
    <Directory /opt/web-apps/web2py>
```

```
#    AllowOverride None
```

```
#    Order Allow,Deny
```

```
#    Deny from all
```

```
#    <Files wsgihandler.py>
```

```
#        Allow from all
```

```
#    </Files>
```

```
    AllowOverride None
```

```
    Require all denied
```

```
    <Files wsgihandler.py>
```

```
        Require all granted
```

```
    </Files>
```

```
</Directory>
```

```
    AliasMatch ^/([^/]+)/static/(?:_[\d]+\.[\d]+\.[\d]+)/?(.*) /  
    opt/web-apps/web2py/applications/$1/static/$2
```

```
    <Directory /opt/web-apps/web2py/applications/*/static>
```

```
#    Options -Indexes
```

```
#    Order Allow,Deny
```

```
#    Allow from all
```

```
    Options -Indexes
```

```
    ExpiresActive On
```

```
    ExpiresDefault "access plus 1 hour"
```

```
    Require all granted
```

```
</Directory>
```

```
<Location /admin>
```

```
    Deny from all
```

```
</Location>
```

```
<LocationMatch ^/([^/]+)/appadmin>
```

```
    Deny from all
```

```
</LocationMatch>
```

```
    CustomLog /var/log/httpd/access_log common
    ErrorLog /var/log/httpd/error_log
</VirtualHost>
```

```
<VirtualHost *:443>
    SSLEngine on
    SSLCertificateFile /etc/httpd/ssl/self_signed.cert
    SSLCertificateKeyFile /etc/httpd/ssl/self_signed.key

    WSGIProcessGroup web2py
    WSGIScriptAlias / /opt/web-apps/web2py/wsgihandler.py
    WSGIPassAuthorization On
```

```
    <Directory /opt/web-apps/web2py>
        AllowOverride None
        Require all denied
        <Files wsgihandler.py>
            Require all granted
        </Files>
    </Directory>
```

```
    AliasMatch ^/([^/]+)/static/(?:_[\d]+\.[\d]+\.[\d]+)/?(.*) /
    opt/web-apps/web2py/applications/$1/static/$2
```

```
    <Directory /opt/web-apps/web2py/applications/*/static>
        Options -Indexes
        ExpiresActive On
        ExpiresDefault "access plus 1 hour"
        Require all granted
    </Directory>
```

```
    CustomLog /var/log/httpd/access_log common
    ErrorLog /var/log/httpd/error_log
</VirtualHost>
```