Université Libre de Bruxelles
Service de Bioinformatique des Génomes et Reséaux (BiGRe)
Laboratory of Genome and Network Biology
http://www.bigre.ulb.ac.be/

Regulatory Sequence Analysis Tools (*RSAT*)
Web server configuration for *RSAT*

Jacques VAN HELDEN & the *RSAT* team

September 30, 2016

Contents

1 Web server configuration for *RSAT* **3**

1.1 Description . . . 3

1.2 Installing a local web server . . . 3

1.2.1 Web server pages . . . 3

1.2.2 Apache modules . . . 3

1.3 Configuring and activating a local ***RSAT***Web server . . . 4

1.3.1 Testing the web server . . . 5

1.4 Managing a local web server . . . 5

1.4.1 Access logs . . . 5

1.4.2 Cleaning the temporary directory . . . 5

1 Web server configuration for *RSAT*

1.1 Description

This documents describes the installation procedure for the web server of the **Regulatory Sequence Analysis Tools** (***RSAT***).

It assumes that you already installed the perl scripts and the genomes, as described in the ***RSAT***installation guide.

1.2 Installing a local web server

The Regulatory Sequence Analysis Tools include a web server, which offers a user-friendly interface for biologists. The main server is available for academic users at
`http://www.rsat.eu/`

Some additional mirrors have been installed in different countries.

1.2.1 Web server pages

The web pages are located in the directory *rsat/public_html*, which contains both the HTML pages, and the CGI scripts.

1.2.2 Apache modules

The ***RSAT***interface relies on CGI (for the earlier tools) and PHP (for the most recent tools). These modules should be installed on the web server, and activated in the Apache configuration files.

Log in as super-user of your server, open the main apache configuration (*/etc/apache2/httpd.conf*) witha text editor, and check that the following lines are uncommented (suppress the leading # if any).

```
LoadModule cgi_module libexec/apache2/mod_cgi.so
LoadModule php5_module libexec/apache2/libphp5.so
```

For Apache version 2.2, you also need to activate the perl module *mod_perl.so*, whereas in version 2.4 this module does not appear anymore (Perl seems to be included in the server).

Uncomment the following line:

```
AddHandler cgi-script .cgi
```

Optional : it is convenient to associate a plain/text mime type to extensions for some classical bioinformatics files, in order to display them directly in the browser rather than proposing users to download them.

In the config file *http.conf*, locate the place with the AddType directives, and add the following lines.

```
AddType text/plain .fasta
AddType text/plain .bed
```

For the Network Analysis Tools, you also need to adapt the PHP parameters in order to support larger data sizes for Web forms (post) and file uploads.

After having found the php.ini file for your operating system [1], modify the following parameters:

```
upload_max_size=100M
post_max_size = 100M
```

1.3 Configuring and activating a local *RSAT* Web server

In order to provide web access to the Regulatory Sequence Analysis Tools (***RSAT***), you need to adapt the configuration of your web server. This requires root privileges (can be done only by the system administrator of the computer).

1. A default configuration file is provided with the ***RSAT*** distribution (*rsat_apache_default.conf*).

   Copy this template to a file named *rsat.conf*, which you will edit to replace the string `[RSAT_PARENT_PATH]` by the full path of your *rsat* folder.

2. The configuration file should then be copied to some appropriate place in the Apache configuration folder of your computer. This place depends on the operating system (Mac OSX or Linux) and on the distribution (Linux Ubuntu, Centos, ...).

   Some Usual places:

   - On Centos: */etc/httpd/conf.d/rsat.conf*
   - On Ubuntu: */etc/apache2/sites-enabled/rsat.conf*
   - On Mac OSX: */etc/apache2/users/rsat.conf*

3. You need to restart the Web server (note: the command depends on your OS. Can be ***apachectl***, ***apache2ctl*** or ***httpd***.

[1]

- Ubuntu: */etc/php5/apache2/php.ini.*
- Mac OSX Yosemite: copy the file */etc/php.ini.default* to */etc/php.ini* and edit this copy.

```
sudo apachectl restart
```

4. Check that all properties related to the Web site URL are properly defined in the ***RSAT*** property files *$RSAT/RSAT_config.props* and *$RSAT/RSAT_config.mk*.

   In principle you already configured these files in the beginning of the installation, with the command

   ```
   perl perl-scripts/configure_rsat.pl
   ```

   Note: it is important to properly define the URL fo the Web server (`RSAT_WWW` and related variables). The default URL (http://localhost/rsat/) only works if the server and client (your Web browser) are on the same machine. This internal access is very convenient to work in places where you don't have Internet connections, but does not allow other computers to use your Web server. If you want to enable Web queries from remote computers, you should specify an externally visible URL.

1.3.1 Testing the web server

To test the web server, open a web browser and connect your ***RSAT*** server (of course you need to adapt the following URL according to the actual IP address of the server).

`http://www.myserver.mydomain/rsat/`

If the connection works, try to execute the demonstration of the following pages.

supported organisms to check that genomes have been insalled.

retrieve-seq to test the correct installation of genomes.

oligo-analysis to test the correct installation of background oligonucleotide frequencies.

feature-map to test the correct installation of the graphical librairies.

1.4 Managing a local web server

1.4.1 Access logs

Each time a script is executed via the ***RSAT*** server, some basic information is stored in a log file. This information is minimal: it is restricted to the time, name of the script executed, and the IP address of the client machine. We do not want to store any additional information (e.g. selected organism, lists of genes), for obvious confidentiality reasons.

The log files are saved in the directory *$RSAT/logs*. There is one file per month.

1.4.2 Cleaning the temporary directory

The web server stores result files in a temporary directory *$RSAT/public_html/tmp/*. These files should remain 3 days on the server, in order to allow users to consult their results.

Manual cleaning

The ***RSAT***package includes a make script to clean old files in the temporary directory.

```
cd $RSAT
make -f makefiles/server.mk clean_tmp
```

This command cleans all the files older than 3 days. You can clean more recent files by modifying the variable CLEAN_DATE.

```
make -f makefiles/server.mk clean_tmp  CLEAN_DATE=1
```

This will clean all files older than 1 day.

Automatic cleaning

The automatic management of the temporary directory can be greatly facilitated the ***crontab*** command. For this, you need to add a command to your personal `crontab` configuration file.

1. Start to edit the crontab command file

   ```
   crontab -e
   ```

   This will open your *crontab* file with your default text editor (this default editor can be specified with the environment variable EDITOR or VISUAL).

2. Add the following line to the *crontab* file.

   ```
   02 04 * * * make -f [RSAT_PARENT_PATH]/rsat/makefiles/server.mk clean_tmp
   ```

   This will execute the make script *server.mk*, with the target `clean_tmp`, every day, at 04:02 AM.

3. Save the modified crontab file and close your text editor.

In principle, you will receive an email from ***crontab*** each time the command is executed.
Note that the command ***crontab*** takes effect only if the system administrator has activated the command

```
cron
```

. If you notice that the temporary files are not properly cleaned, please contact your system administrator to check the cron command.