

Regulatory Sequence Analysis Tools

Web server installation

Jacques van Helden

jvanheld@scmbb.ulb.ac.be

<http://www.scmbb.ulb.ac.be/~jvanheld/>

Service de Conformation des Macromolécules Biologiques et de Bioinformatique,

Université Libre de Bruxelles,

Campus Plaine, CP 263, Boulevard du Triomphe, B-1050 Bruxelles, Belgium.

Tel: +32 2 650 2013 - Fax: +32 2 650 5425

August 25, 2005

Contents

1	Description	3
2	Installing a local web server	3
2.1	Web server pages	3
2.2	Configuration of the Apache server	3
2.3	Editing the configuration file for the <i>RSAT</i> server	4
2.4	Configuring the RSAT options for the web server	4
2.4.1	IP address of your machine	4
2.4.2	Setting the environment variable RSAT	5
2.5	Testing the web server	5
3	Managing a local web server	5
3.1	Access logs	5
3.2	Cleaning the temporary directory	5
3.2.1	Manual cleaning	6
3.2.2	Automatic cleaning	6

1 Description

This document 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.

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://rsat.ulb.ac.be/rsat/>. A few additional mirrors have been installed in different countries.

2.1 Web server pages

The web pages are located in the directory *rsa-tools/public_html*. This directory contains both the HTML pages, and the CGI scripts.

2.2 Configuration of the Apache server

In order to provide web access to the *rsa-tools*, you need to adapt the configuration of your web server. This requires root privileges (system administrator).

The detailed settings depend on your web server program. We provide here an example of typical settings for the Apache server (the most widely used web server).

In summary, the configuration includes the following steps.

1. Open the apache configuration file */etc/httpd/httpd.conf*
2. Specify an alias for *rsa-tools* in the appropriate section.

```
Alias /rsa-tools "/home/myaccount/rsa-tools/public_html/"
```

3. Associate *.cgi* extension to CGI scripts

Make sure the following line is present in the config file. If the server has not yet been configured, the line is commented, and you need to remove the # character before it.

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

4. Give authorization to execute CGI scripts in the *rsa-tools* directory

```
ScriptAlias /rsa-tools/ "/home/myaccount/rsa-tools/public_html/"
```

5. Specify the access options for the *rsa-tools* directory.

```
<Directory "/home/myaccount/rsa-tools/public_html">
    AllowOverride None
    Options ExecCGI Indexes
    Order allow,deny
    Allow from all
</Directory>
```

These are the basic steps to configure the web access to **RSAT**. Depending on your operating system, you probably need to specify some additional settings. For example, on the Max OSX version of Apache server allows to define a user-specific configuration in the directory `/etc/httpd/users`.

Note that you need to restart the web server for these changes to take effect. The command to restart the server depends on its version. You can try the following commands, and if they don't work check the documentation of your web server.

```
httpd restart
```

```
apachectl restart
```

2.3 Editing the configuration file for the *RSAT* server

If you want to install a web server, you need to edit two variables on the file *RSA.config*. Open this file with a text editor, and specify the variables `$config_site` and `$WWW_RSA` according to your local configuration.

2.4 Configuring the RSAT options for the web server

You need to adapt the RSAT configuration file to indicate the configuration of your server. For this, edit the file `$RSAT/RSA.config`.

2.4.1 IP address of your machine

By default, the RSAT web server is configured to be used from the machine on which it is installed.

```
$WWW_RSA = "http://localhost/rsa-tools/";
```

If you want to give access from external machines, you need to adapt the following line and replace "localhost" by the IP address of your machine. Let us assume that your machine has the address `www.myserver`. You should then edit the row as follows.

```
$WWW_RSA = "http://www.myserver/rsa-tools/";
```

2.4.2 Setting the environment variable RSAT

For the basic RSAT configuration, you had to define an environment variable RSAT. This is necessary for the perl program to know the location of your programs. If this variable is not specified, the scripts are stopped with an error message.

```
unless ($ENV{RSAT}) {  
    die "Error: the environment variable RSAT needs to be defined\n";  
}
```

Since the web server is ran by another user, you need to define this environment variable for this user as well. For this, edit the above line in the following way.

```
unless ($ENV{RSAT}) {  
    $ENV{RSAT} = "/home/myaccount/rsa-tools";  
}
```

2.5 Testing the web server

To test the web server, open a web browser and connect your **RSAT** server

http://www.myserver/rsa-tools/

Of course you need to adapt the URL according to your IP address.

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

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 libraries.

3 Managing a local web server

3.1 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 *rsa-tools/logs*. There is one file per month.

3.2 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.

3.2.1 Manual cleaning

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

```
cd rsa-tools
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.

3.2.2 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 /home/myaccount/rsa-tools/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.