

# rtorrent+rutorrent en contenedor

## instalacion rtorrent+rutorrent

1. crear estructura de directorios:

```
mkdir -p data/downloads/completo
mkdir -p data/config
```

2. copiar ficheros de configuración (extraídos de propio contenedor y modificados a conveniencia):

- [rutorrent-tls.nginx](#)

```
server {
    listen 443 ssl;
    listen [::]:443 default_server ipv6only=on;

    keepalive_timeout 60;
    ssl_certificate /etc/nginx/ssl/nginx.crt;
    ssl_certificate_key /etc/nginx/ssl/nginx.key;
    ssl_ciphers "AES128+EECDH:AES128+EDH";
    ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
    ssl_prefer_server_ciphers on;
    ssl_session_cache shared:SSL:10m;
    add_header X-Frame-Options SAMEORIGIN;
    add_header X-Content-Type-Options nosniff;

    root /var/www/rutorrent;
    index index.php index.html index.htm;

    # Make site accessible from http://localhost/
    server_name localhost;

    location / {
        # First attempt to serve request as file, then
        # as directory, then fall back to displaying a 404.
        try_files $uri $uri/ =404;
        # Uncomment to enable naxsi on this location
        # include /etc/nginx/naxsi.rules
        auth_basic "Restricted";
        auth_basic_user_file /var/www/rutorrent/.htpasswd;
    }

    # Only for nginx-naxsi used with nginx-naxsi-ui : process denied
    requests
    #location /RequestDenied {
    #    proxy_pass http://127.0.0.1:8080;
    #}

    #error_page 404 /404.html;

    # redirect server error pages to the static page /50x.html
    #
```

```
#error_page 500 502 503 504 /50x.html;
#location = /50x.html {
#    root /usr/share/nginx/html;
#}

# pass the PHP scripts to FastCGI server listening on
127.0.0.1:9000
#
location ~ /\.php$ {
    fastcgi_split_path_info ^(.+\.(php|php5))(/.+)$;
# NOTE: You should have "cgi.fix_pathinfo = 0;" in php.ini

# With php5-cgi alone:
#    fastcgi_pass 127.0.0.1:9000;
# # With php5-fpm:
fastcgi_pass unix:/var/run/php/php7.2-fpm.sock;
fastcgi_index index.php;
include fastcgi_params;
    fastcgi_param HTTPS on;
    fastcgi_param SCRIPT_FILENAME $request_filename;
}

# deny access to .htaccess files, if Apache's document root
# concurs with nginx's one
#
#location ~ /\.ht {
#    deny all;
#}

location /RPC2 {
    include scgi_params;
    scgi_pass 127.0.0.1:5000;
    scgi_param SCRIPT_NAME /RPC2;
}

location /completo {

try_files $uri $uri/ =404;
# Uncomment to enable naxsi on this location
# include /etc/nginx/naxsi.rules
    auth_basic "Restricted";
    auth_basic_user_file /var/www/rutorrent/.htpasswd;

    alias /downloads/completo;
    autoindex on;
}
}
```

- copiar fichero **config.php** en **data/config** (para cambiar variable \$forbidUserSettings = true; - originalmente en false, evita creación de perfiles por usuario .htaccess)

3. crear enlaces simbólicos:

```
# los enlaces funcionarán desde dentro del contenedor por el mapeo de volumen
ln -s /etc/letsencrypt/live/k1.vividumcodex.com/cert.pem nginx.crt
ln -s /etc/letsencrypt/live/k1.vividumcodex.com/privkey.pem nginx.pem
```

4. ejecutar:

```
docker run \
  -dt \
  --name rtorrent-rutorrent \
  -p 443:443 \
  -p 49160:49160/udp \
  -p 49161:49161 \
  -v $PWD/data/downloads:/downloads \
  -v /etc/letsencrypt:/etc/letsencrypt \
  -v $PWD/data/config/config.php:/var/www/rutorrent/conf/config.php \
  -v $PWD/data/config/rtorrent-tls.nginx:/root/rtorrent-tls.nginx \
  diameter/rtorrent-rutorrent:latest
```

## old

- [rtorrent+rutorrent en contenedor \(OLD1\)](#)
- [rtorrent+rutorrent en contenedor \(OLD2\)](#)

## + info

- <https://www.rapidseedbox.com/kb/beginners-guide-to-rtorrent>
- <https://github.com/Novik/ruTorrent/wiki/Config>

From:  
<https://miguelangel.torresegea.es/wiki/> - **miguel angel torres egea**

Permanent link:  
[https://miguelangel.torresegea.es/wiki/linux:debian:seedbox:rtorrent\\_rutorrent?rev=1608802907](https://miguelangel.torresegea.es/wiki/linux:debian:seedbox:rtorrent_rutorrent?rev=1608802907)

Last update: **24/12/2020 01:41**

