

Установка и настройка rtorrent+rutorrent+nginx+php-fpm в Arch Linux

Предыстория

Для работы с торрентами я достаточно долго использовал ktorrent. Сей клиент мои потребности в удобном управлении закачками удовлетворял полностью, пока я не заметил, что на популярных торрентах загрузка процессора доходила до 50% (а с uTP — ещё больше), а потребление памяти в и без того жирном KDE стало неприятно бросаться в глаза.

Было решено сменить KDE на xfce (это отдельная история), а программу для торрентов подобрать с хорошим функционалом и удобным управлением. Перепробовав transmission, deluge и rtorrent, я остановился на последнем.

О том, как настроить rtorrent+rutorrent+nginx+php-fpm, и будет под катом.

Почему так?

Сразу хочу ответить, почему выбраны именно означенные инструменты.

Arch Linux. Об этом дистрибутиве можно говорить достаточно долго, мне нравится его организация и философия в целом, и я могу его использовать для себя с максимальной эффективностью. Кому интересно более детально прочитать о нём, смотрите [сюда](#) и [сюда](#).

nginx. Я впечатлён тем, как эта маленькая штука выполняет свою работу, экономя память, гибко настраиваясь и предоставляя весь необходимый мне функционал.

php-fpm. Можно настроить количество рабочих потоков, в связке с nginx даёт

замечательную производительность.

rtorrent. Малое потребление ресурсов, хорошо настраивается.

rutorrent. Активно развивается, имеет приятный интерфейс.

Установка необходимого программного обеспечения

Будем исходить из того, что Arch Linux на компьютере уже стоит, а пользователь знаком с его пакетной системой.

Чтобы установить веб-часть связки, выполняем команду:

```
sudo pacman -S nginx php-fpm
```

rtorrent и libtorrent я рекомендую устанавливать с AUR'a, там есть замечательный PKGBUILD под названием rtorrent-color, делающий скучный консольный интерфейс более приятным (*если будете им пользоваться*), и libtorrent-extended, имеющий дополнительные патчи. Поэтому выполняем команду:

```
yaourt rtorrent-color
```

и

```
yaourt libtorrent-extended
```

Чтобы получить rutorrent, нужно клонировать его из svn'a командой:

```
svn checkout rutorrent.googlecode.com/svn/trunk/ rutorrent-read-only
```

Файлы появятся в каталоге «rutorrent-read-only», потом мы их оттуда заберём.

Настройка

В файле /etc/php/php-fpm.conf нужно установить такие параметры:

- **listen = 127.0.0.1:9000**, чтобы php-fpm слушал на указанном сетевом сокете;
- **pm = static**, чтобы количество рабочих потоков было постоянным;
- **pm.max_children = 2**, чтобы установить количество рабочих потоков, равным количеству физических потоков (*у меня двухъядерный процессор, поэтому здесь установлено 2*).

Файл /etc/nginx/conf/nginx.conf приведём к такому виду:

```
worker_processes 2;

error_log /var/log/nginx/error.log;
pid /var/run/nginx.pid;

events {
worker_connections 1024;
use epoll;
}
```

```
http {
include /etc/nginx/conf/mime.types;
default_type application/octet-stream;

upstream backend {
server 127.0.0.1:9000;
}

upstream backendrtorrent {
server unix:/home/pf/.rtorrent.sock;
}

sendfile on;
keepalive_timeout 65;

include /etc/nginx/conf/sites-enabled/*;
include /etc/nginx/conf/conf.d/*;
}
```

Обратите внимание, что подсекция **backend** указывает на php-fpm, а **backendrtorrent** — на socket-файл rtorrent (об этом далее).

Создадим каталоги **/etc/nginx/conf/sites-enabled** и **/etc/nginx/conf/sites-available**. Во втором создадим конфигурационный файл **rtorrent.eternity** следующего содержания и сделаем на него символическую ссылку в первом каталоге:

```
server {
listen 80;
server_name localhost;

access_log /srv/http/nginx/rtorrent.eternity/logs/access.log;
error_log /srv/http/nginx/rtorrent.eternity/logs/errors.log;

location / {
root /srv/http/nginx/rtorrent.eternity/htdocs;
index index.php index.html index.htm;
}

location /RPC2 {
include /etc/nginx/conf/scgi_params;
```

```
scgi_pass backendrtorrent;
}

location ~ /\.ht {
deny all;
}

location ~*\.(jpg|jpeg|gif|bmp|ico|png|css|js|swf)$ {
root /srv/http/nginx/rutorrent.eternity/htdocs;
access_log off;
expires 30d;
}

location ~ .php$ {
fastcgi_split_path_info ^(.+\.(php)).*$;
fastcgi_pass backend;
fastcgi_index index.php;
fastcgi_param SCRIPT_FILENAME /srv/http/nginx/rutorrent.eternity/htdocs$fastcgi_script_name;
include fastcgi_params;
fastcgi_param QUERY_STRING $query_string;
fastcgi_param REQUEST_METHOD $request_method;
fastcgi_param CONTENT_TYPE $content_type;
fastcgi_param CONTENT_LENGTH $content_length;
fastcgi_intercept_errors on;
fastcgi_ignore_client_abort off;
fastcgi_connect_timeout 60;
fastcgi_send_timeout 180;
fastcgi_read_timeout 180;
fastcgi_buffer_size 128k;
fastcgi_buffers 4 256k;
fastcgi_busy_buffers_size 256k;
fastcgi_temp_file_write_size 256k;
}
}
```

Обратите внимание на местоположение журналов ошибок и доступа (**/srv/http/nginx/rutorrent.eternity/logs**), а также на корневой каталог веб-документов (**/srv/http/nginx/rutorrent.eternity/htdocs**). Эти каталоги должны существовать. Также обратите внимание на блок **location /RPC2**, он обязателен для rutorrent. Описание остальных параметров здесь я не привожу, подробной документации в Интернете достаточно.

В файл **/etc/hosts** внесите такую строчку:

```
127.0.0.1 localhost.localdomain localhost eternity rutorrent.eternity
```

eternity — название моей системы, у вас оно может быть другим. В таком случае его нужно поменять везде в конфигурационных файлах.

Можно запускать nginx и php-fpm:

```
sudo /etc/rc.d/nginx start
```

```
sudo /etc/rc.d/php-fpm start
```

Можно теперь протестировать веб-часть связки, закинув в корневой каталог веб-документов какой-нибудь простенький php-файл. При переходе в веб-браузере по адресу rutorrent.eternity он должен корректно отобразиться.

Теперь нужно установить rutorrent. Перенесите из вышеупомянутого каталога **rutorrent-read-only/rtorrent** его содержимое в корневой каталог веб-документов (*напомню, это каталог /srv/http/nginx/rutorrent.eternity/htdocs*). То же сделайте с каталогом **rutorrent-read-only/plugins**, его скопируйте поверх существующего каталога plugins в дереве файлов rutorrent. Мусор вида **.svn** можно удалить.

Откройте файл **/srv/http/nginx/rutorrent.eternity/htdocs/conf/config.php** и замените там всего две строчки:

```
$scgi_port = 0;
```

```
$scgi_host = "unix:///home/pf/.rtorrent.sock";
```

Сокет-файл должен совпадать с упомянутым выше.

Веб-интерфейс готов, теперь нужно настроить сам rtorrent.

Создайте в домашнем каталоге файл .rtorrent.rc с таким содержимым:

```
scgi_local = /home/pf/.rtorrent.sock
max_memory_usage = 268435456
system.file_allocate.set = yes
done_fg_color = 2
done_bg_color = 0
active_fg_color = 4
active_bg_color = 0
download_rate = 250
upload_rate = 250
directory = /home/pf/work/downloads/torrents
session = /home/pf/work/downloads/torrents/.session
port_range = 29292-29292
check_hash = no
use_udp_trackers = yes
encryption = allow_incoming,try_outgoing,enable_retry,prefer_plaintext
dht = auto
dht_port = 6881
peer_exchange = yes
```

Строчка **system.file_allocate.set = yes** имеет смысл, если libtorrent скомпилирован с опцией `--with-posix-fallocate`, что на современных ФС даёт возможность моментально выделить нужное пространство под торрент. Опции вида ***g_color** относятся только к `rtorrent-color`. Каталоги, порты и скорость настройте на своё усмотрение.

Последний штрих — скрипт для запуска `rtorrent`. Следующее содержимое поместите в файл `/etc/rc.d/rtorrentd`:

```
#!/usr/bin/env bash

. /etc/rc.conf
. /etc/rc.d/functions

rtorrent_user="pf"
rtorrent_socket="/home/pf/.rtorrent.sock"

case "$1" in
start)
stat_busy "Starting rtorrent"
if [ -S $rtorrent_socket ]; then
rm $rtorrent_socket
fi
su $rtorrent_user -c 'LANG=uk_UA.UTF-8 screen -d -m -S rtorrent rtorrent' &> /dev/null
if [ $? -gt 0 ]; then
stat_fail
else
while [ ! -S $rtorrent_socket ]
do
printf "%10s\r" waiting
done
chmod 666 $rtorrent_socket
add_daemon rtorrent
stat_done
fi
;;
stop)
stat_busy "Stopping rtorrent"
killall -w -s 2 /usr/bin/rtorrent &> /dev/null
if [ -S $rtorrent_socket ]; then
rm $rtorrent_socket
fi
if [ $? -gt 0 ]; then
stat_fail
```



```
else
rm_daemon rtorrent
stat_done
fi
;;
restart)
$0 stop
sleep 1
$0 start
;;
*)
echo "usage: $0 {start|stop|restart}"
esac
exit 0
```

Естественно, должен быть установлен screen. Переменные **rtorrent_user** и **rtorrent_socket** поправьте под своё окружение. Если нужна автозагрузка rtorrent, поместите его в файл **/etc/rc.conf** в массив **DAEMONS**.

Всё. Запускайте rtorrent командой

```
sudo /etc/rc.d/rtorrentd start
```

заходите в браузере на сайт rtorrent.eternity и наслаждайтесь.

pfactum, 29 мая в 00:47