Transmission - торрент-клиент для FreeBSD с web-интерфейсом

Опубликовано muff в Пнд, 2013-09-30 15:43



Поскольку все чаще и чаще необходимые файлы можно обнаружить только в **peer-to-peer** сетях, решил на одном из ненагруженых серверов установить **torrent-клиент**, который работал бы круглосуточно. Выбор остановил на **transmission**, поскольку управление даным **torrent**-клиентом можно выполнять через веб-интерфейс.

Выполним установку из системы портов:

```
# cd /usr/ports/net-p2p/transmission-daemon && make install clean && rehash
```

Внесем необходимые опции запуска в **rc.conf**:

```
# echo '# Запуск transmission' >> /etc/rc.conf
# echo 'transmission_enable="YES"' >> /etc/rc.conf
# echo '# Каталог для .torrent-файлов' >> /etc/rc.conf
# echo 'transmission_watch_dir="/usr/torrents/.torrents/"' >> /etc/rc.conf
# echo '# Каталог для загруженых файлов' >> /etc/rc.conf
# echo 'transmission_download_dir="/usr/torrents/downloads/"' >> /etc/rc.conf
# echo '# Каталог конфигурационных файлов' >> /etc/rc.conf
# echo 'transmission_conf_dir="/usr/torrents/.transmission/"' >> /etc/rc.conf
# echo '# Имя пользователя, от имени которого будет работать transmission' >>
/etc/rc.conf
# echo 'transmission_user="transmission"' >> /etc/rc.conf
```

Если опция запуска **transmission_user** не указана, то процесс запускается под пользователем **transmission**, а если задана пустой - тогда под **root**, что не рекомендуется. Также есть возможность указать дополнительные флаги запуска, воспользовавшись опцией запуска **transmission_flags=""**.

Следующий шаг - создание каталогов и установка необходимых прав доступа. В процессе установки будет создан пользователь **transmission**. Если же хотите использовать другого пользователя - необходимо будет предварительно его создать.

```
# mkdir -p /usr/torrents/.torrents
# mkdir /usr/torrents/downloads
# mkdir /usr/torrents/.transmission
# chown -R transmission:transmission /usr/torrents
```

Пора приступать к созданию конфигурационного файла. Самый простой способ - запуск и останов **transmission**. В процессе запуска он проверит наличие необходимых файлов и в

случае необходимости, создаст их.

```
# sh /usr/local/etc/rc.d/transmission start
# sh /usr/local/etc/rc.d/transmission stop
```

После этого приступаем к редактированию конфигурационного файла **settings.json**, который будет создан в каталоге **transmission_conf_dir**. Немного подкорректировал дефолтный файл конфигурации. В результате имеем следующие параметры:

```
{    "alt-speed-down": 50,    "alt-speed-enabled": false,    "alt-speed-time-begin":
 540,    "alt-speed-time-day": 127,    "alt-speed-time-enabled": false,    "alt-spee
d-time-end": 1020,    "alt-speed-up": 50,    "bind-address-ipv4": "10.227.206.45
",    "bind-address-ipv6": "::",    "blocklist-enabled": false,    "blocklist-url":
"http://www.example.com/blocklist [1]
",  "cache-size-mb": 4,  "dht-enabled": true,  "download-dir": "/usr/torrents/download
s/",  "download-queue-enabled": true,  "download-queue-size": 5,  "encryption": 1,  "
idle-seeding-limit": 30,  "idle-seeding-limit-enabled": false,  "incomplete-dir": "/usr/tor
rents/.incomplete",  "incomplete-dir-enabled": true,  "lpd-enabled": false,  "message-l
evel": 2,  "peer-congestion-algorithm": "",  "peer-id-ttl-hours": 6,  "peer-limit-global":
200,  "peer-limit-per-torrent": 50,  "peer-port": 51413,  "peer-port-random-high": 599
99,    "peer-port-random-low": 50000
,    "peer-port-random-on-start": false,    "peer-socket-tos": "default",    "pex-enabled": true,    "pidfil
e": "/var/run/transmission/daemon.pid",    "port-forwarding-enabled": true,    "preallocation": 1,    "pr
efetch-enabled": 1,    "queue-stalled-enabled": true,    "queue-stalled-minutes": 30,    "ratio-limit": 2,
    "ratio-limit-enabled": false,    "rename-partial-files": true,    "rpc-authentication-required": true
,    "rpc-bind-address": "10.227.206.45",    "rpc-enabled": true
,    "rpc-password": "very_hard_pawssword",    "rpc-port": 9095
,    "rpc-url": "/transmission/",    "rpc-username": "muff",    "rpc-whitelist": "127.0.0.1,192.168.*.*
",    "rpc-whitelist-enabled": true,    "scrape-paused-torrents-enabled": true,    "script-torrent-done-e
nabled": false,    "script-torrent-done-filename": "",    "seed-queue-enabled": false,    "seed-queue-siz
e": 10,    "speed-limit-down": 100,    "speed-limit-down-enabled": false,    "speed-limit-up": 100,    "s
peed-limit-up-enabled": false,    "start-added-torrents": true,    "trash-original-torrent-files": false,    "
umask": 18,    "upload-slots-per-torrent": 14,    "utp-enabled": true,    "watch-dir": "/usr/torrents/.torr
ents/",    "watch-dir-enabled": true}
```

Значительная часть опций понятна и без комментариев. Рассмотрим некоторых из опций.

- **alt-speed-enabled** - альтернативные настройки скорости;
- **alt-speed-time-begin** - время в минутах от полуночи до начала действия альтернативных правил;
- **alt-speed-time-day** - битовая маска дней, по которым действуют правила
    - Sunday: 1 (binary: 0000001)
    - Monday: 2 (binary: 0000010)
    - Tuesday: 4 (binary: 0000100)
    - Wednesday: 8 (binary: 0001000)
    - Thursday: 16 (binary: 0010000)
    - Friday: 32 (binary: 0100000)
    - Saturday: 64 (binary: 1000000)
      Например:
    - Рабочие дни: 62 (binary: 0111110)
    - Выходные: 65 (binary: 1000001)
    - Все дни: 127 (binary: 1111111)

- **alt-speed-time-enabled** - альтернативные настройки скорости в зависимости от времени суток;
- **alt-speed-time-end** - время в минутах от полуночи до окончания действия альтернативных правил;
- **bind-address-ipv4** - на каком IPv4 адресе "биндить" демон;

- **rpc-authentication-required** - обязательна ли аутентификация для подключившихся;
- **rpc-bind-address** - на каком IPv4 адресе "биндить" удаленный доступ;
- **rpc-enabled** - включение удаленного доступа;
- **rpc-password** - пароль для подключения; хешируется после рестарта;
- **rpc-port** - на каком порту "слушать" запросы к **RPC**
- **rpc-url** - по какому адресу ожидать запросов;
- **rpc-username** - имя пользователя для подключения;
- **rpc-whitelist** - перечень IP-адресов, разделенных запятой, для которых разрешены удаленные подключения;
- **rpc-whitelist-enabled** - включать ли rpc-whitelist

Попытаемся запустить **transmission**:

```
# sh /usr/local/etc/rc.d/transmission start
```

Сервис успешно стартовал. Если же нет - ищем в логах причину ошибки. Также после успешного старта можно обнаружить такие записи в **/var/log/messages**:

```
... UDP Failed to set send buffer: No buffer space available ...
... UDP Failed to set receive buffer: No buffer space available ...
```

Исправить эту ситуацию можно увеличив максимально допустимый размер приемного буфера операционной системы:

```
# echo 'kern.ipc.maxsockbuf=2097152' >> /etc/sysctl.conf
# sysctl kern.ipc.maxsockbuf=2097152
```

Либо же, как вариант, можно в **settings.json** переменную **utp-enabled** выставить в значение "**false**".

ВАЖНО! В случае необходимости внесения изменений в файл конфигурации **settings.json**, предварительно необходимо остановить **transmission**.

После запуска сервиса можно заходить на веб-интерфейс, используя логин и пароль, указанный в конфигурационном файле. Пройдя авторизацию, нам доступен следующий интерфейс:



[2]

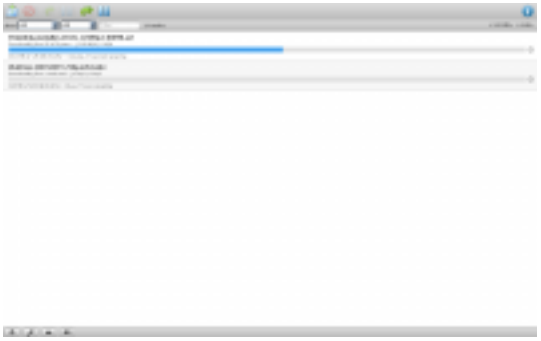
Добавим торрент-файл, воспользовавшись иконкой "**Open Torrent**" в верхнем левом углу интерфейса:



[3]

После добавления торрента загрузка будет запущена автоматически:



[4] 

[5]

С остальными возможностями предоставленного интерфейса разобраться не составит труда.

Кстати, **transmission** дает довольно большое поле для творчества, разрешая запуск внешних скриптов.

Доступные переменные:

- **$TR_APP_VERSION** - версия **transmission**
- **$TR_TORRENT_ID** - id торрента, как оно показывается в remote-GUI
- **$TR_TORRENT_NAME** - имя торрента как оно отображается в интерфейсе
- **$TR_TORRENT_DIR** - директория торрента
- **$TR_TORRENT_HASH** - хэш торрента
- **$TR_TIME_LOCALTIME** - дата и время запуска стрипта

Например, сделаем так, что по завершению закачки на почту будет приходить уведомление. Для этого напишем скрипт **finish.sh** следующего содержания:

```
#!/bin/sh

echo $TR_TIME_LOCALTIME - $TR_TORRENT_NAME Downloaded | mail -s "Torrent download successfull" root [at] domain [dot] com (root [at] domain [dot] com)

exit 0;
```

Остановим **transmission** и внесем необходимые изменения в **settings.json**:

```
"script-torrent-done-enabled": true,
"script-torrent-done-filename": "/usr/torrents/finish.sh",
```

После этого снова запускаем **transmission**.

Источник (получено *2026-11-12 17:58*):
http://muff.kiev.ua/content/transmission-torrent-klient-dlya-freebsd-c-web-interfeisom

Ссылки:
[1] http://www.example.com/blocklist
[2] http://muff.kiev.ua/files/imagepicker/1/transmission-03.png
[3] http://muff.kiev.ua/files/imagepicker/1/transmission-01.png
[4] http://muff.kiev.ua/files/imagepicker/1/transmission-00.png
[5] http://muff.kiev.ua/files/imagepicker/1/transmission-02.png