configuración rtorrent

- instalar subversion: `apt-get install subversion`
- instalar compilador: `apt-get instsall make`
- crear carpeta 'source' en 'user'
- descargar:
  - ~~svn co~~ https://xmlrpc-c.svn.sourceforge.net/svnroot/xmlrpc-c/stable ~~xmlrpc~~
    - svn co http://svn.code.sf.net/p/xmlrpc-c/code/advanced xmlrpc-c
    - svn co http://svn.code.sf.net/p/xmlrpc-c/code/stable xmlrpc-c
  - ~~curl~~ http://libtorrent.rakshasa.no/downloads/libtorrent-0.13.3.tar.gz ~~| tar xz~~
    - curl http://rakshasa.github.io/rtorrent/libtorrent-0.13.4.tar.gz | tar xz
  - ~~curl~~ http://libtorrent.rakshasa.no/downloads/rtorrent-0.9.3.tar.gz ~~| tar xz~~
    - curl http://rakshasa.github.io/rtorrent/rtorrent-0.9.4.tar.gz | tar xz
- compilamos el xmlrpc
  - `cd xmlrpc`
  - `./configure –prefix=/usr –enable-libxml2-backend –disable-libwww-client –disable-wininet-client –disable-abyss-server –disable-cgi-server`
  - `sudo make`
  - `sudo make install`
- compilamos el paquete libtorrent
  - `sudo ./configure –prefix=/usr`
  - `sudo make -j2`
  - `sudo make install`
- compilamos el paquete rtorrent
  - `export LD_LIBRARY_PATH=/usr/lib`
  - `sudo ./autogen.sh`
  - `sudo ./configure –prefix=/usr –with-xmlrpc-c`
  - `sudo make -j2`
  - `sudo make install`
- `sudo ldconfig`
- creamos el fichero de configuración del rtorrent, en el directorio home del usuario

```
# This is an example resource file for rTorrent. Copy to
# ~/.rtorrent.rc and enable/modify the options as needed. Remember to
# uncomment the options you wish to enable.
#
# Based on original .rtorrent.rc file from http://libtorrent.rakshasa.no/
# Modified by Lemonberry for rtGui http://rtgui.googlecode.com/
#
# This assumes the following directory structure:
#
# /Torrents/Downloading - temporaray location for torrents while downloading (see
"directory")
# /Torrents/Complete - Torrents are moved here when complete (see "on_finished")
# /Torrents/TorrentFiles/Auto - The 'autoload' directory for rtorrent to use. Place
a file
# in here, and rtorrent loads it #automatically. (see "schedule = watch_directory")
# /Torrents/Downloading/rtorrent.session - for storing rtorrent session information
#
```

```
# Maximum and minimum number of peers to connect to per torrent.
#min_peers = 40
max_peers = 100

# Same as above but for seeding completed torrents (-1 = same as downloading)
min_peers_seed = -1
max_peers_seed = -1

# Maximum number of simultanious uploads per torrent.
max_uploads = 50

# Global upload and download rate in KiB. "0" for unlimited.
download_rate = 0
upload_rate = 0

# Default directory to save the downloaded torrents.
directory = /home/user/downloading

# Default session directory. Make sure you don't run multiple instance
# of rtorrent using the same session directory. Perhaps using a
# relative path?
session = /home/user/.session

# Watch a directory for new torrents, and stop those that have been
# deleted.
schedule = watch_directory,5,5,load_start=/home/usuario/torrent/*.torrent
schedule = untied_directory,5,5,stop_untied=

# Close torrents when diskspace is low. */
schedule = low_diskspace,5,60,close_low_diskspace=100M

# Stop torrents when reaching upload ratio in percent,
# when also reaching total upload in bytes, or when
# reaching final upload ratio in percent.
# example: stop at ratio 2.0 with at least 200 MB uploaded, or else ratio 20.0
#schedule = ratio,60,60,stop_on_ratio=200,200M,2000


# When the torrent finishes, it executes "mv -n <base_path> ~/Download/"
# and then sets the destination directory to "~/Download/". (0.7.7+)
#on_finished = move_complete,"execute=mv,-u,$d.get_base_path=,/home/user/complete/
;d.set_directory=/home/user/complete/"
system.method.set_key = event.download.finished,move_complete,"execute=mv,-
u,$d.get_base_path=,/home/user/complete/ ;d.set_directory=/home/user/complete/"
# The ip address reported to the tracker.
#ip = 127.0.0.1
#ip = rakshasa.no

# The ip address the listening socket and outgoing connections is
# bound to.
#bind = 127.0.0.1
#bind = rakshasa.no

# Port range to use for listening.
port_range = 55995-56000
```

```
# Start opening ports at a random position within the port range.
#port_random = yes

scgi_port = 127.0.0.1:5000

# Check hash for finished torrents. Might be usefull until the bug is
# fixed that causes lack of diskspace not to be properly reported.
#check_hash = no

# Set whetever the client should try to connect to UDP trackers.
#use_udp_trackers = no

# Alternative calls to bind and ip that should handle dynamic ip's.
#schedule = ip_tick,0,1800,ip=rakshasa
#schedule = bind_tick,0,1800,bind=rakshasa

# Encryption options, set to none (default) or any combination of the following:
# allow_incoming, try_outgoing, require, require_RC4, enable_retry,
prefer_plaintext
#
# The example value allows incoming encrypted connections, starts unencrypted
# outgoing connections but retries with encryption if they fail, preferring
# plaintext to RC4 encryption after the encrypted handshake
#
encryption = allow_incoming,enable_retry,prefer_plaintext

# Enable DHT support for trackerless torrents or when all trackers are down.
# May be set to "disable" (completely disable DHT), "off" (do not start DHT),
# "auto" (start and stop DHT as needed), or "on" (start DHT immediately).
# The default is "off". For DHT to work, a session directory must be defined.
#
dht = disable

# UDP port to use for DHT.
#
# dht_port = 6881

# Enable peer exchange (for torrents not marked private)
#
peer_exchange = no

#
# Do not modify the following parameters unless you know what you're doing.
#

# Example of scheduling commands: Switch between two ip's every 5
# seconds.
#schedule = "ip_tick1,5,10,ip=torretta"
#schedule = "ip_tick2,10,10,ip=lampedusa"

# Remove a scheduled event.
#schedule_remove = "ip_tick1"
```