

# Running a storage node

## Overview

A lobber storage node is essentially a standard BitTorrent client with a very thin layer used to receive notification about new and modified torrents. A lobber storage node is provided as a twisted plugin and is installed from the python package index. Currently the twisted plugin works with `transmission` - a popular, cross-platform BitTorrent client. Other BT clients may be supported in the future.

## Installing

The examples below assume a debian/ubuntu environment. If you're running another unix-like or windows OS your milage may vary. Our goal is for lobber to run on a multitude of environments and operating systems. Please provide us with patches!

### Transmission

We recommend installing transmission 2.03 or later. If you are using ubuntu 10.04 you may have to install transmission from a PPA:

```
# apt-get install python-software-properties
# add-apt-repository ppa:transmissionbt/ppa
# apt-get update
```

If you run Ubuntu 10.10 you onlu need to do the next bit.

```
# apt-get install transmission-cli transmission-daemon deluge-common
```

At this point you should have a transmission daemon running after each reboot. Use the `/etc/init.d/transmission-daemon` tool to stop/start the service in the normal way.

### lobberstoragenode

The lobberstoragenode is available from <http://pypi.python.org/> and can be installed using the `easy_install` tool from python-setuptools:



Errata: You may have to download the tarball and install it using the provided `setup.py` instead of using `easy_install`.

```
# apt-get install python-setuptools python-twisted python-feedparser bittorrent
# easy_install stompservice transmissionrpc
```

```
# easy_install lobberstoragenode
```

or install lobberstoragenode from GIT or from tarballs...

At this point `twistd` should have a new plugin registered. Verify this by running `twistd --help` and look for `lobberstoragenode` in the list of commands. If you see this then you're installation should be ok.

```

# twistd --help
...
Commands:
  ftp                An FTP server.
  telnet             A simple, telnet-based remote debugging service.
  socks             A SOCKSv4 proxy service.
  manhole-old       An interactive remote debugger service.
  portforward       A simple port-forwarder.
  lobberstoragenode A Storage Node for Lobber
  web               A general-purpose web server which can serve from a
                  filesystem or application resource.
  inetd            An inetd(8) replacement.
  news             A news server.
  xmpp-router       An XMPP Router server
  words            A modern words server
  toc              An AIM TOC service.
  dns              A domain name server.
  mail            An email service
  manhole          An interactive remote debugger service accessible via
                  telnet and ssh and providing syntax coloring and
                  basic line editing functionality.
  conch            A Conch SSH service.

```

Running `twistd lobberstoragenode --help` gives you an overview of the options:

```

$ twistd lobberstoragenode --help
Usage: twistd [options] lobberstoragenode [options]
Options:
  -n, --standardNotifications  Add standard notification destinations
  -R, --register                Register new torrents with lobber
  -a, --announceUrl=           Announce URL (tracker) to use for new
                              torrents
  -A, --acl=                   Access Control List to apply to new torrents
  -b, --dropbox=               A directory to watch for new content
  -d, --torrentDir=            The directory where to store torrents
                              [default: torrents]
  -D, --transmissionDownloadsDir= The downloads directory for transmission
                              [default:
                              /var/lib/transmission-daemon/downloads]
  -h, --lobberHost=            The host running both STOMP and https for
                              lobber
  -k, --lobberKey=              The Lobber application key to use
  -p, --trackerProxyTrackerUrl= Enable tracker proxying for given https
                              tracker (HOST[:PORT])
  -P, --trackerProxyListenOn=  Adress to bind the tracker proxy to [default:
                              localhost:8080]
  -r, --removeLimit=           Remove torrent and data when this many other
                              storage-nodes have the data (0=never remove)
                              [default: 0]
  -S, --stompUrl=              The STOMP protocol URL to use for
                              notifications [default:
                              stomp://localhost:61613]
  -T, --transmissionRpc=        The RPC URL for transmission [default:
                              http://transmission:transmission@localhost:9091]
  --version
  --help                        Display this help and exit.

```

When running a storage-node for development purposes you will typically run it with both the `--stompUrl` and `--lobberUrl` parameters pointing to endpoints on your local development sandbox. When running a storage-node off of `beta.lobber.se` you will typically want to start it this way:

```
# twistd -n lobberstoragenode -k <appkey> -h beta.lobber.se -d /var/torrents [<destination|rss>]*
```

The final arguments is a set of STOMP destinations and/or RSS feeds. Each RSS feed is pulled regularly and each link is downloaded. If the downloaded data is a BitTorrent file it is stored in the `--torrentDir` directory (`/var/torrents` in the example above). Optionally if a `--script` option was given it will be run as a command with the path to the torrent-file as an argument.

If you want to notify transmission of new torrents you might do something like this

```
# twistd -n lobberstoragenode ... --script="transmission-remote -n transmission:transmission -a"
```

A more complete example. This would regularly pull all torrents linked from <https://beta.lobber.se/torrent/tag/foo.rss> and ask transmission to add the new ones.

```
# twistd -n lobberstoragenode -k <key> -h beta.lobber.se -d /tmp \\  
--script="transmission-remote -n transmission:transmission -a" \\  
https://beta.lobber.se/torrent/tag/foo.rss /torrent/tag/foo
```

## Useful stuff to listen to for a storagenode

### STOMP destinations

Location	What it gets you
/torrent/new	Each newly created torrent
/torrent/tag/add	Each time a torrent is tagged
/torrent/tag/add/foo	Each time a torrent is tagged with 'foo'
/torrent/tag/remove/foo	Each time the 'foo' tag is removed from a torrent

### RSS URLs

Location	What it gets you
<a href="https://lobber.example.org/torrents/all.rss">https://lobber.example.org/torrents/all.rss</a>	All torrents readable by the current user (or key)
<a href="https://lobber.example.org/torrents/tag/foo.rss">https://lobber.example.org/torrents/tag/foo.rss</a>	All torrents tagged with 'foo'