Running a storage node (bud light)

Page index

1. Overview
2. Installing on Ubuntu
3. Installing on OS X
4. Configuration
    a. Stand-alone configuration
    b. Lobber web app configuration
    c. OS X specific configuration

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.

The examples below assume a debian/ubuntu and OS X 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!

Installing on Ubuntu

Install 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
# apt-get install transmission-cli transmission-daemon
```

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.

Install lobber-storagenode

Install dependencies:

```
apt-get install python-pip python-setuptools python-twisted
```

```
pip install stompservice transmissionrpc feedparser
```

Download the tar archive lobber-storagenode-2.tar.gz or get the storage node from git.nordu.net:

```
git clone git://git.nordu.net/lobber-storagenode.git
```

Install the storagenode:

```
(for tar archive) tar xvzf lobber-storagenode-2-tar.gz
cd lobber-storagenode
python setup.py install
```

Installing on OS X

Install Transmission

Install Transmission for OS X and enable "Remote Access" in Preferences. We recommend installing transmission 2.03 or later.

Install lobber-storagenode

Download the tar archive lobber-storagenode-2.tar.gz or get the storage node from git.nordu.net:

```
git clone git://git.nordu.net/lobber-storagenode.git
```

Configuration

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 notificiation 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
  -T, --transmissionRpc=           The RPC URL for transmission
      --version
      --help                       Display this help and exit.
```

Stand-alone storage node

First you will have to create a key with the right entitlements to be able to access the torrents you are interested in. You do this in the Lobber web app under Application Keys.

Find the configuration file config in the directory /etc/lobberstoragenode/.

You need to change the following options:

lobberstoragnode

```
L_MODE="standalone"

L_KEY="[KEY]"
L_HOST="tracker.someserver.net"

L_TRACKERPROXYURL="https://tracker.someserver.net:443/tracker/uannounce"
L_TRACKERPROXYLISTEN="localhost:8080"
L_URLS="https://tracker.someserver.net/torrent/all.json ... ..."
```

L_URLS is a set of STOMP destinations and/or RSS feeds. Each destination is pulled regularly and each link is downloaded. If the downloaded data is a BitTorrent file it is stored in the `--torrentDir` directory and added to transmission.

By default (L_STDNOTIFY="True") the standalone storage node will listen to stomp://tracker.someserver.net/torrent/new.

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 torrrent

RSS URLs

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

Start the storage node:

```
/etc/init.d/lobberstoragenode start
```

To make the storagenode start automatically start on reboot run:

```
update-rc.d lobberstoragenode defaults
```

Storage node for Lobber web application

To start a storage node to complement the Lobber web application you need first to create a key with the right entitlements and url filter via the web ui.

To make the storage node able to write in transmissions standard download directory you need to add the group debian-transmission to the www-data user.

```
usermod -G debian-transmission www-data
```

Find the configuration file config in the directory /etc/lobberstoragenode/.

You need to change the following options:

lobberstoragnode

```
L_MODE="webapp"

L_KEY="[KEY]"
L_HOST="localhost"
L_DROPBOXDIR="/path/to/lobber/dropbox"
```

The `L_KEY` is the lobber application key you obtain from lobber. The `L_HOST` should be 'beta.lobber.se' in order to use the Lobber Beta site and `L_DROPBOXDIR` is the path to a directory in your file system that will act as a dropbox. Any file or directory placed here will be uploaded to lobber and then removed.

Start the storage node:

```
/etc/init.d/lobberstoragenode start
```

To make the storagenode start automatically start on reboot run:

```
update-rc.d lobberstoragenode defaults
```

OS X specific configuration

These are the changes that are needed to make the storage node work with the OS X version of Transmission.

```
L_TORRENTDIR="/Users/[username]/Downloads"
L_TRANSMISSIONDLDIR="/Users/[username]/Downloads"
L_TRANSMISSIONRPC="http://localhost:9091"
L_USER="[username]"
L_GROUP_PERM="staff"
```