
ProjectTox Documentation

Release 0.1

Tox Team

August 02, 2013

CONTENTS

Contents:

USING TOX

Note: There is a German version of this page available: `start_guide.de`

1. *Build Tox*
2. Fix errors
3. Consult IRC for help
4. Go on debugging journey for devs
5. Build Tox for real
6. ???

For all the work we’ve put into Tox so far, there isn’t yet a decent guide for how you *use* Tox. Here’s a user-friendly attempt at it.

1. Connect to the network!

- You need to connect to a bootstrapping server, to give you a public key.
- Where can I find a public server? Right here, as of now: (the help message from running `nTox` with no args will help)

```
- 198.46.136.167 33445 728925473812C7AAC482BE7250BCCAD0B8CB9F737BF3D42ABD34459C170
- 192.81.133.111 33445 8CD5A9BF0A6CE358BA36F7A653F99FA6B258FF756E490F52C1F98CC420F
- 66.175.223.88 33445 AC4112C975240CAD260BB2FCD134266521FAAF0A5D159C5FD3201196191E
- 192.184.81.118 33445 5CD7EB176C19A2FD840406CD56177BB8E75587BB366F7BB3004B19E3ED0
```

2. Find a friend!

- Now that you’re on the network, you need a friend. To get one of those, you need to send or receive a request. What’s a request, you ask? It’s like a friend request, but we use really scary and cryptic numbers instead of names. When `nTox` starts, it shows *your* long, scary number, called your *public key*. Give that to people, and they can add you as a “friend”. Or, you can add someone else, with the `/f` command, if you like.

3. Chat it up!

- Now use the `/m` command to send a message to someone. Wow, you’re chatting!

4. But something broke!

- Yeah, pre-alpha-alpha software tends to do that. We’re working on it.
- Please report all crashes to either the GitHub page, or `#tox-dev` on freenode.

5. Nothing broke, but what does `/f` mean?

- `nTox` parses text as a command if the first character is a forward-slash (`/`). You can check all commands in `commands.md`.

6. Use and support Tox!

- Code for us, debug for us, document for us, translate for us, even just talk about us!
- The more interest we get, the more work gets done, the better Tox is.

INSTALL INSTRUCTIONS

2.1 Linux

First, install the build dependencies

```
bash apt-get install build-essential libtool autotools-dev automake libconfig-dev ncurses-dev cmake
```

Note: libconfig-dev should be ≥ 1.4 .

Then you'll need a recent version of [libsodium](#)

```
git clone git://github.com/jedisct1/libsodium.git
cd libsodium
git checkout tags/0.4.2
./autogen.sh
./configure && make check
sudo checkinstall --install --pkgname libsodium --pkgversion 0.4.2 --nodoc
sudo ldconfig``
```

Finally, fetch the Tox source code and run cmake

```
git clone git://github.com/irungentoo/ProjectTox-Core.git
cd ProjectTox-Core && mkdir build && cd build
cmake ..
```

Then you can build any of the files in [/testing](#) and [/other](#) that are currently supported on your platform by running

```
make name_of_c_file
```

For example, to build [Messenger_test.c](#) you would run

```
make Messenger_test
```

Or you could just build everything that is supported on your platform by running

```
bash make
```

2.2 OS X

2.2.1 Homebrew

```
brew install libtool automake autoconf libconfig libsodium cmake
cmake .
make
sudo make install
```

2.2.2 Non-homebrew

Much the same as Linux, remember to install the latest XCode and the developer tools (Preferences -> Downloads -> Command Line Tools). Users running Mountain Lion and the latest version of XCode (4.6.3) will also need to install libtool, automake and autoconf. They are easy enough to install, grab them from <http://www.gnu.org/software/libtool/>, <http://www.gnu.org/software/autoconf/> and <http://www.gnu.org/software/automake/>, then follow these steps for each:

```
./configure
make
sudo make install
```

Do not install them from macports (or any dependencies for that matter) as they get shoved in the wrong directory and make your life more annoying.

Another thing you may want to install is the latest gcc, this caused me a few problems as XCode from 4.3 no longer includes gcc and instead uses LLVM-GCC, a nice install guide can be found at <http://caiustheory.com/install-gcc-421-apple-build-56663-with-xcode-42>

2.3 Windows

You should install:

- [MinGW's C compiler](#)
- [CMake](#)

You have to [modify your PATH environment variable](#) so that it contains MinGW's bin folder path. With default settings, the bin folder is located at `C:\MinGW\bin`, which means that you would have to append `;C:\MinGW\bin` to the PATH variable.

Then you should either clone this repo by using git, or just download a [zip of current Master branch](#) and extract it somewhere.

After that you should get precompiled package of libsodium from [here](#) and extract the archive into this repo's root. That is, sodium folder should be along with core, testing and other folders.

Navigate in cmd to this repo and run:

```
mkdir build && cd build
cmake -G "MinGW Makefiles" ..
```

Then you can build any of the [/testing](#) and [/other](#) that are currently supported on your platform by running:

```
mingw32-make name_of_c_file
```

For example, to build [Messenger_test.c](#) you would run:

```
mingw32-make Messenger_test ``
```

Or you could just build everything that is supported on your platform by running:

```
mingw32-make
```


TOX USER COMMANDS

Here's a list of commands that nTox accepts, which can all be used by starting your line with a /. Currently there can be no spaces before this.

- `/f [ID]`
 - Add a friend with ID [ID].
- `/d`
 - Call `doMessenger()` which does...something?
- `/m [FRIEND_NUM] [MESSAGE]`
 - Message [FRIEND_NUM] [MESSAGE].
- `/n [NAME]`
 - Change your username to [NAME].
- `/l`
 - Print your list of friends. (like you have any)
- `/s [STATUS]`
 - Set your status to [STATUS].
- `/a [ID]`
 - Accept friend request from [ID].
- `/i`
 - Print useful info about your client.
- `/h`
 - Print some help.
- `/q/`
 - Quit Tox. (why ;_;))

INDICES AND TABLES

- *genindex*
- *modindex*
- *search*