

Contents

- [1 Introduction to Ubuntu and FreeSWITCH](#)
- [2 Initial Setup](#)
- [3 Install from Repo](#)
 - ◆ [3.1 Create Ubuntu packages and install them](#)
 - ◆ [3.2 Old method for building packages](#)
- [4 Build from source](#)
 - ◆ [4.1 Install dependencies](#)
 - ◆ [4.2 Build and install with makefiles](#)
- [5 Problem info from the quick and dirty page](#)
- [6 Using FreeTDM with the 'direct from git' method above](#)
- [7 Issues with specific Ubuntu releases](#)
 - ◆ [7.1 Ubuntu 10.04 LTS 64bit](#)
 - ◆ [7.2 Ubuntu 11.10 64bit](#)

Introduction to Ubuntu and FreeSWITCH

This page was intended to help those who want to install FreeSWITCH, and not start learning how FreeSWITCH actually works. This will give you the basic setup so that you can at least start making calls.

Please also review the entries for Debian and Ubuntu under [http://wiki.freeswitch.org/wiki/Download %26 Installation Guide#Debian](http://wiki.freeswitch.org/wiki/Download_%26_Installation_Guide#Debian) for some more information before proceeding.

Initial Setup

The instructions below have been updated for 11.04+.

Note that in the command line examples, the editors Nano (command name "nano") or Vim (Vi Improved, command name "vim") are mentioned. Any text editor (gedit, emacs, etc.) should be able to be substituted. Often this command is in the EDITOR environment variable (referenced with \$EDITOR).

Install the standard Ubuntu server system.

Install from Repo

There are packages in the repo for distributions up to Maverick. If you're running a later version (which you probably are...), you should build freeswitch manually or create deb packages and install them (see next section).

Open a terminal window (if necessary) and then type these commands:

```
apt-get install python-software-properties
add-apt-repository ppa:freeswitch-drivers/freeswitch-nightly-drivers
```

Ubuntu_Quick_Start

```
apt-get update
apt-get install freeswitch freeswitch-lang-en freeswitch-sounds-en-us-callie-8000
```

Following the above will get you a basic installation of FreeSWITCH installed in /opt/freeswitch.

Create Ubuntu packages and install them

This script will download everything required, build packages and installs them. It works on a basic installation. Save it as freeswitch_build.sh in your home directory and start it with

```
sh -x freeswitch_build.sh
```

It's (always) recommended to build as a regular user, NOT root. Tested on Ubuntu 12.04 LTS, with Freeswitch 1.2.9 (which means separate music and sounds package). Partly based on the Debian installer script.

It will give a permission error when starting freeswitch, disable spandsp in /etc/freeswitch/autoload_configs/modules.conf to avoid this, didn't look into the problem as of yet. You can run Freeswitch as root also.

```
#!/bin/bash
BUILDDIR=~/.src
BRANCH=v1.2.stable
DCH_DISTRO=UNRELEASED
sudo apt-get -y update
sudo apt-get -y install autoconf automake devscripts gawk g++ git-core libjpeg-dev libncurses5-dev
sudo apt-get -y -f install
sudo update-alternatives --set awk /usr/bin/gawk
mkdir $BUILDDIR
cd $BUILDDIR
git clone -b $BRANCH git://git.freeswitch.org/freeswitch.git
DISTRO=`lsb_release -cs`
FS_VERSION="$(cat freeswitch/build/next-release.txt | sed -e 's/-/~/'~n$(date +%Y%m%dT%H%M%SZ)-

(cd freeswitch && build/set-fs-version.sh "$FS_VERSION")
(cd freeswitch && dch -b -m -v "$FS_VERSION" --force-distribution -D "$DCH_DISTRO" "Custom build.
if [ -f modules.conf ]; then cp modules.conf freeswitch/debian; fi
(cd freeswitch/debian && ./bootstrap.sh -c ${DISTRO})
sudo mk-build-deps -i freeswitch/debian/control
(cd freeswitch && dpkg-buildpackage -b -uc)

mkdir dbg
mv *dbg* dbg

cd $BUILDDIR
git clone https://github.com/traviscross/freeswitch-sounds.git
cd freeswitch-sounds
./debian/bootstrap.sh -p freeswitch-sounds-en-us-callie
./debian/rules get-orig-source
tar -xv --strip-components=1 -f *_*.orig.tar.xz && mv *_*.orig.tar.xz ../
dpkg-buildpackage -uc -us -Zxz -z9

cd $BUILDDIR
rm -rf freeswitch-sounds
git clone https://github.com/traviscross/freeswitch-sounds.git
cd freeswitch-sounds
./debian/bootstrap.sh -p freeswitch-music-default
```

Ubuntu_Quick_Start

```
./debian/rules get-orig-source
tar -xv --strip-components=1 -f *_*.orig.tar.xz && mv *_*.orig.tar.xz ../
dpkg-buildpackage -uc -us -Zxz -z9

sudo sed -i "$ a\deb file:${BUILDDIR} ./" /etc/apt/sources.list
cd $BUILDDIR
sudo dpkg-scanpackages . /dev/null | gzip -9c > Packages.gz
sudo apt-get -y update
sudo apt-get -y --force-yes install freeswitch-meta-vanilla freeswitch-music
sudo mkdir /etc/freeswitch
sudo cp -r /usr/share/freeswitch/conf/vanilla/* /etc/freeswitch/
sudo adduser --disabled-password --quiet --system --home /usr/share/freeswitch --gecos "FreeSwitch"
sudo sh -c 'chown -R freeswitch:daemon /etc/freeswitch/'
sudo sh -c 'chown -R freeswitch:daemon /usr/share/freeswitch/'
#fix music paths
cd /usr/share/freeswitch/sounds/music
sudo ln -s default/8000 8000
sudo ln -s default/16000 16000
sudo ln -s default/32000 32000
#done!!
cd $BUILDDIR
sudo /etc/init.d/freeswitch start
```

Old method for building packages

```
#Make sure you installed all dependencies above, and that you checked out the freeswitch code in
#
#To select which modules you want to create packages for (there will be a deb for each module at
#create a modules.conf in the freeswitch debian directory. If you want a copy of modules.conf to
#bootstrap script in the debian directory, then copy the generated modules_.conf to modules.conf
#Run the bootstrap script again afterwards.

cd /usr/src/freeswitch/debian
./bootstrap.sh

#CHECK FOR BUILD DEPENDENCIES
cd ..

# method 1)
dpkg-checkbuilddeps
# if the above comes back with anything, install the packages with apt-get.

# method 2)
# (first make sure that you've installed devscripts and equivs with apt-get)
# this has the benefit of making it easier to uninstall all the build dependencies:
mk-build-deps -i

#then:
dpkg-buildpackage -b

#This will take quite some time. Once it is done, lets see all the debs that got created:
cd /usr/src
ls freeswitch*.deb

#There most likely will be a crap load of files there. Lets move the dbg packages to another dir
mkdir dbg
mv *dbg_*.deb dbg
```

Ubuntu_Quick_Start

```
#Then install the freeswitch lib package first
dpkg -i libfreeswitch1_*.deb

#Then install the rest of the deb packages
dpkg -i freeswitch*.deb

#If the above command gives you problems about packages not being configured, this is most likely

#You will then need to create your own configuration directory in /etc/freeswitch.
cp /usr/src/freeswitch/conf/vanilla /etc/freeswitch

#That's it!
/etc/init.d/freeswitch start

#wait for freeswitch to start and then connect to it:
fs_cli
```

Also, make sure to check out `debian/README.source` and `debian/README.Debian` as those are updated when significant changes are made to the debian build or setup process.

Build from source

It's recommended to build Ubuntu packages..

Install dependencies

Open a terminal window (if necessary) and then type these commands:

```
cd /usr/src
sudo apt-get install git-core subversion build-essential autoconf automake libtool libncurses5 libncurses5-dev
# optional packages
# sudo apt-get install libcurl4-openssl-dev libexpat1-dev libgnutls-dev libtiff4-dev libx11-dev uclibc-dev
                                zlib1g-dev libzrtpp-dev libasound2-dev libogg-dev libvorbis-dev libperl5.10-dev
                                uuid-dev

#for ubuntu 11.04+
sudo apt-get install gawk
update-alternatives --set awk /usr/bin/gawk

git clone git://git.freeswitch.org/freeswitch.git
cd freeswitch
```

Build and install with makefiles

Run the following in the freeswitch directory:

```
./bootstrap.sh
./configure
make
sudo make install
sudo make uhd-sounds-install
sudo make uhd-moh-install
```

Old method for building packages

Ubuntu_Quick_Start

```
sudo make samples
```

```
#Following the above resulted in a default install in /usr/local/ not /opt, so you must do the fo  
sed /usr/src/freeswitch.git/debian/freeswitch-sysvinit.freeswitch.init -e s,opt,usr/local, >/etc/  
chmod 755 /etc/init.d/freeswitch
```

```
#make the init.d/freeswitch script start at boot  
update-rc.d -f freeswitch defaults
```

```
cp debian/freeswitch-sysvinit.freeswitch.default /etc/default/freeswitch # (** change to "true" *
```

```
#Add the freeswitch user:
```

```
adduser --disabled-password --quiet --system \  
  --home /opt/freeswitch \  
  --gecos "FreeSwitch Voice Platform" --ingroup daemon \  
  freeswitch
```

```
adduser freeswitch audio
```

```
#Change freeswitch installation ownership (from root)  
chown -R freeswitch:daemon /opt/freeswitch/
```

```
#Remove permissions for other  
chmod -R o-rwx /opt/freeswitch/
```

```
#Add the fs_cli to your default path  
ln -s /opt/freeswitch/bin/fs_cli /usr/local/bin/
```

```
#Then either:  
sudo shutdown -r now
```

```
#or  
/etc/init.d/freeswitch start
```

```
# test its startup  
fs_cli
```

Problem info from the quick and dirty page

The quick and dirty page is now generic and this info was recent and for Ubuntu:

```
# for Ubuntu 10.04/10.10, to get Dingaling working, you need to use the trick  
# mentioned here: http://lists.freeswitch.org/pipermail/freeswitch-users/2010-July/060357.html  
# not sure whether this is FreeSWITCH configure issue or Ubuntu issue.**)
```

Using FreeTDM with the 'direct from git' method above

earthspike, 2011-06-05

The /dev/wan* are owned by root.root, so this needs to change for mod_freetdm to work correctly.

I got mine to work (on 10.04.2 LTS) by:

- Create user and group freeswitch as described above. Command for adding the group is

Ubuntu_Quick_Start

```
addgroup --system freeswitch
```

- Change the group ownership on the devices:

```
chgrp freeswitch /dev/wan*
```

- Restart freeswitch if already running:

```
/etc/init.d/freeswitch restart
```

- Use fs_cli to check that your TDM spans are showing:

```
ftdm list
```

Issues with specific Ubuntu releases

Ubuntu 10.04 LTS 64bit

Tested Dec 23, 2010 with Ubuntu

There were no issues during installation and testing.

Ubuntu 11.10 64bit

Tested Oct 14, 2011 with Ubuntu

Tested Oct 26, 2011 with Ubuntu

The default awk is now mawk and not gawk. Gawk must be installed to compile successfully. [Changes in the default GCC toolchain behavior](#) may require other [tweaks like the ones described here](#).

For the latest git head (2011/10/26), libz and pkg-config, are dependencies that were required to build properly. libjpeg-dev is required for spandsp

```
apt-get install git-core build-essential autoconf automake libtool libncurses5 libncurses5-dev g
```

Note: If you receive errors in libs/iksemel complaining of an expected ")" this may point to the pkg-config package not being installed. After installing pkg-config you will need to bootstrap the environment again to expand the proper aclocal macros.

To resolve the dependency on gawk, the system awk link was updated to point to gawk using the following command

```
update-alternatives --set awk /usr/bin/gawk
```