



Running Remote Frontends

This chapter discusses how to build remote frontends for your MythTV installation. Up until now, we have shown how to run the MythTV frontend on the same machine as the backend, but you don't have to set up your MythTV installation that way. In fact, you can have more than one frontend if you want. You might want a remote frontend for a variety of reasons. Perhaps your backend machine is too big or too loud to be in your living room. Perhaps you want to be playing more than one video at once in different parts of the house.

This chapter will cover building three types of remote frontends. First we'll talk briefly about Linux-based frontends, then we'll talk about Xbox-based frontends (which are a slightly more specialized version of the generic Linux frontend), and finally we'll cover Macintosh frontends. We'll also briefly cover options for viewing videos on machines other than your backend without going the whole way and installing an entire frontend.

Getting Ready for Other Frontends

You need to take some preparatory steps before you can run frontends on machines other than the machine on which the backend runs. First, you need to configure the network to have the MythTV backend machine on the same static IP address all the time. Second, you need to configure the MythTV backend to be accessible to the network. Finally, you need to tweak the MySQL configuration to work on the network as well. If you change the MythTV network configuration and find that it's still not working, check your firewall settings.

Network Configuration

Obviously, you need a network connection between your MythTV backend machine and the various frontend machines on your network. You probably already have your backend machine on the network so you can get TV guide data information from the Internet, so this shouldn't need a lot of work. However, it is also important that you have a static IP address for your backend machine.

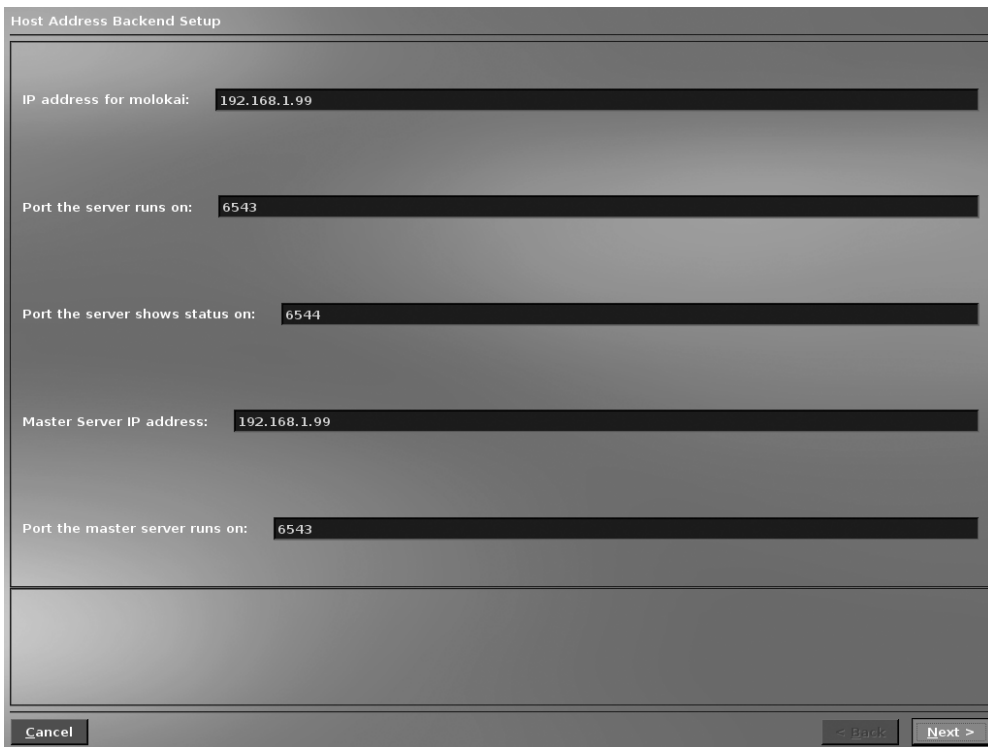
It's hard to describe how to configure your network connection for the backend machine to have a static IP address in a book like this. You can use a variety of command-line tools, or you can use graphical tools, which of course vary depending on what window manager you are using. Your best bet is to check out *Beginning Ubuntu Linux: From Novice to Professional, Second Edition* (Apress, 2007) or to search online for descriptions of how to set the static IP for your network card. You should remember to ensure that you use an IP address that your DHCP

server (on small home LANs, this is usually on your router) isn't handing out, though, because otherwise you will end up with two machines fighting over that address.

Backend Configuration

You first need to configure the backend software to use an IP address that is accessible from the network (it defaults to 127.0.0.1 or localhost, which is not network accessible). You do this by running `mythtv-setup` and going to the general preferences. The first page of that wizard looks like Figure 8-1.

Tip You might find that you don't have a `mythtv-setup` program. On some distributions, it is named `mythtvsetup`.



Host Address Backend Setup

IP address for molokai: 192.168.1.99

Port the server runs on: 6543

Port the server shows status on: 6544

Master Server IP address: 192.168.1.99

Port the master server runs on: 6543

Cancel < Back Next >

Figure 8-1. *Setting the IP address of the backend*

You can see here that the IP address of this backend is 192.168.1.99. Yours has probably defaulted to 127.0.0.1 and should be changed to the IP address of the machine running the backend. You can find out the IP address of the machine that is running the backend by going to a command prompt and running the `ifconfig` command. In this example, we know that

the name of the interface is eth0; if you don't know the name of the interface, then just omit that portion of the command line:

```
$ ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:13:20:9D:71:A5
          inet addr:192.168.1.99  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::213:20ff:fe9d:71a5/64  Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:360121 errors:0 dropped:0 overruns:0 frame:0
          TX packets:644094 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:25279027 (24.1 MiB)  TX bytes:935139149 (891.8 MiB)
```

MySQL Configuration

Now, you need to configure MySQL to be accessible on your network. The first step is to tell MySQL to listen on the IP address that is accessible to the network, in the same way you did for the MythTV backend. However, MySQL doesn't have a pretty GUI configuration tool, so you'll need to edit the `/etc/mysql/my.cnf` file yourself. Run your favorite editor to do this, but remember to use `sudo` so that you have root permissions for your edits.

Look for a stanza like this in the configuration file:

```
# Instead of skip-networking the default is now to listen only on
# localhost, which is more compatible and is not less secure.
bind-address            = 192.168.1.99
```

Here you can see that we set the bind address to the same IP address that we used for the MythTV backend in the previous section. You should edit this to be whatever the IP address for your backend is.

Next, you need to give the new frontend permissions to access the MySQL server. You'll find a sample script to do that in the MythTV source code in a directory called `database`. Here is a modified version that does only the bare minimum to get a new frontend working. We've highlighted the parts you'll need to tweak for your setup in bold.

```
GRANT ALL ON mythconverg.* TO mythtv@remoteip IDENTIFIED BY "newpassword";
FLUSH PRIVILEGES;
GRANT CREATE TEMPORARY TABLES ON mythconverg.* TO mythtv@remoteip ➔
IDENTIFIED BY "mythtv";
FLUSH PRIVILEGES;
```

Here you need to replace `remoteip` with the IP address of the machine the new frontend is going to run on or with `'%'` if you either don't care what IP is connecting to MySQL from or are using DHCP. If you're installing a machine specifically to run your new frontend, then you might need to return to this section later when you know the IP address you have allocated to the machine. You also need to specify a remote password (`newpassword`), which the remote machine will use to connect. It's important to note that the password does not have to be the same for each IP, even if the username is the same, which is very cool.

You can get a dump of the users who are allowed to connect to the MySQL database by asking MySQL like this:

```
$ mysql -u root -p mysql
```

```
Enter password: <your superuser password for MySQL>
```

```
Reading table information for completion of table and column names
```

```
You can turn off this feature to get a quicker startup with -A
```

```
Welcome to the MySQL monitor. Commands end with ; or \g.
```

```
Your MySQL connection id is 124 to server version: 5.0.19-Debian_3-log
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
```

```
mysql> select * from user \G;
```

```
***** 1. ROW *****
```

```
...
```

```
***** 2. ROW *****
```

```
Host: 192.168.1.100
```

```
User: mythtv
```

```
Password: 0476fc996afffe24
```

```
Select_priv: N
```

```
Insert_priv: N
```

```
Update_priv: N
```

```
Delete_priv: N
```

```
Create_priv: N
```

```
Drop_priv: N
```

```
Reload_priv: N
```

```
Shutdown_priv: N
```

```
Process_priv: N
```

```
File_priv: N
```

```
Grant_priv: N
```

```
References_priv: N
```

```
Index_priv: N
```

```
Alter_priv: N
```

```
Show_db_priv: N
```

```
Super_priv: N
```

```
Create_tmp_table_priv: N
```

```
Lock_tables_priv: N
```

```
Execute_priv: N
```

```
Repl_slave_priv: N
```

```
Repl_client_priv: N
```

```
ssl_type:
```

```
ssl_cipher:
```

```
x509_issuer:
```

```
x509_subject:
```

```
max_questions: 0
```

```
max_updates: 0
```

```
max_connections: 0
```

```
...
```

We've used the \G option with the SELECT statement, which presents the information in this list format instead of the traditional ASCII art table form, because the data is too wide to fit on a book page or screen. You can see here the details for one of our remote frontends, which has an IP address of 192.168.1.100.

Specific Types of Frontend

Now we can discuss how to set up specific types of frontends. Each of the following discussions is separate, and you can skip the sections you are not interested in without running the risk of ending up with a setup that doesn't work.

Running Another Linux Frontend

You build Linux frontends in the same manner as you built the MythTV machine in Chapters 2 and 3. When you're compiling the MythTV code, you don't need to build the backend as well as the frontend, and the backend on these machines won't be used. Unfortunately, there is no easy way to disable compiling the backend code, so we'll just show how to compile it and then never use the backend.

Frontends don't need their own copies of MySQL running or XMLTV and the guide data setup, so don't install them on the new frontend machine. Here is a brief overview of the relevant steps to build MythTV for your new frontend. You can find a more complete discussion in Chapters 2 and 3.

```
$ sudo apt-get build-dep mythtv
Password:
Reading package lists... Done
Building dependency tree... Done
The following NEW packages will be installed:
...
Need to get 37.9MB of archives.
After unpacking 135MB of additional disk space will be used.
Do you want to continue [Y/n]? y
Get:1 http://security.ubuntu.com dapper-security/main ➤
libfreetype6-dev 2.1.10-1ubuntu2.2 [677kB]
...
$ sudo apt-get install libqt3-mt-mysql
Reading package lists... Done
Building dependency tree... Done
The following NEW packages will be installed:
  libqt3-mt-mysql
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 51.9kB of archives.
After unpacking 139kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com dapper-updates/universe ➤
libqt3-mt-mysql 3:3.3.6-1ubuntu6 [51.9kB]
Fetched 51.9kB in 1s (47.0kB/s)
Selecting previously deselected package libqt3-mt-mysql.
```

```
(Reading database ... 82299 files and directories currently installed.)
Unpacking libqt3-mt-mysql (from ../libqt3-mt-mysql_3%3a3.3.6-1ubuntu6_i386.deb) ...
Setting up libqt3-mt-mysql (3.3.6-1ubuntu6) ...
```

```
$ wget "http://www.mythtv.org/modules.php?name=Downloads&d_op=getit&lid=129"
--13:25:10-- http://www.mythtv.org/modules.php?name=Downloads&d_op=getit&lid=129
=> 'modules.php?name=Downloads&d_op=getit&lid=129'
Resolving www.mythtv.org... 140.211.167.131
Connecting to www.mythtv.org|140.211.167.131|:80... connected.
HTTP request sent, awaiting response... 302 Found
Location: http://ftp.osuosl.org/pub/mythtv/mythtv-0.20.tar.bz2 [following]
--13:25:25-- http://ftp.osuosl.org/pub/mythtv/mythtv-0.20.tar.bz2
=> 'mythtv-0.20.tar.bz2'
Resolving ftp.osuosl.org... 64.50.236.52, 64.50.238.52
Connecting to ftp.osuosl.org|64.50.236.52|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 12,380,677 (12M) [application/x-tar]

100%[=====>] 12,380,677 158.27K/s ETA 00:00

13:44:36 (158.41 KB/s) - 'mythtv-0.20.tar.bz2' saved [12380677/12380677]
```

Notice that this `wget` download redirects to a file named `mythtv-0.20.tar.bz2` and that it will likely be slightly different for newer releases. Now extract the download, and then compile:

```
$ tar --bzip -xf mythtv-0.20.tar.bz2
$ cd mythtv-0.20
$ ./configure
...
$ make
...
$ make install
...
```

After doing all this, you have all the software you need to run a frontend, although you still need to configure the remote control using `lirc`, as discussed in Chapter 3. You might need to add `/usr/local/lib/` to `/etc/ld.so.conf` if you see errors about libraries not being found. This is as simple as editing this file as root and adding the new line. After that, you need to rebuild the library cache with this command:

```
$ sudo ldconfig
```

Now that you have all the software ready, you need to run `mythfrontend` to start configuring the frontend. You'll see a selection screen that asks you to pick your language (see Figure 8-2).

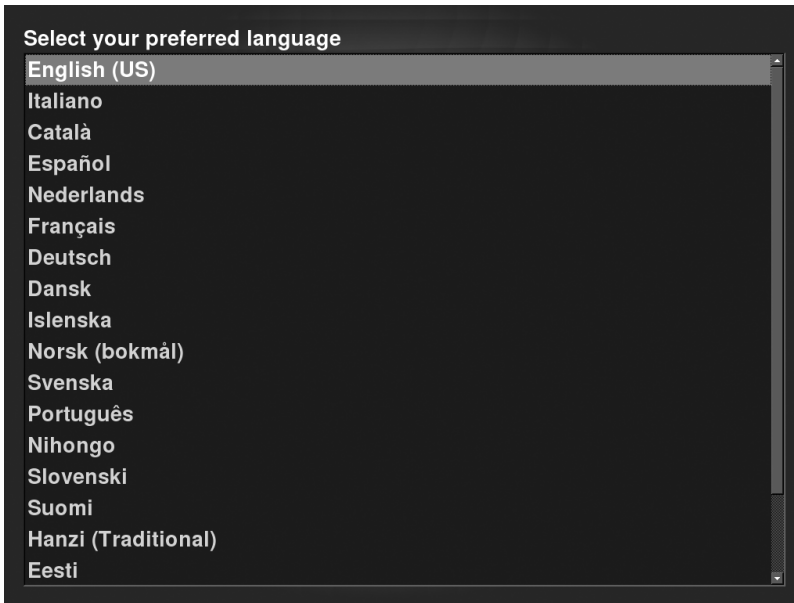


Figure 8-2. *mythtv-setup* prompting for your language choice

Next you're prompted for database configuration (see Figure 8-3).

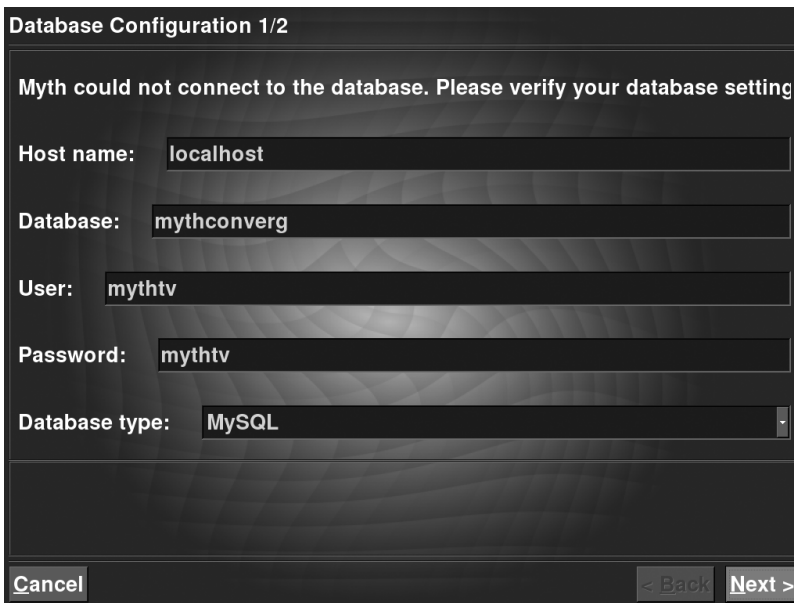


Figure 8-3. *Database configuration defaults*

You need to change these default values to whatever is correct for your network (see Figure 8-4).

Database Configuration 1/2

Myth could not connect to the database. Please verify your database setting

Host name: 192.168.1.99

Database: mythconverg

User: myhtv

Password: xxxxxxxx

Database type: MySQL

The password to use while connecting to the database. This information is required.

Cancel < Back Next >

Figure 8-4. Setting the IP address of the backend, as well as the username and password

Here you need to set up the IP address (or DNS name) of the machine running the backend, the name of the MySQL user who is going to connect from this frontend, and the MySQL password that is valid for that username, connecting from the IP address that the frontend is configured with. (As always, remember that MySQL usernames are not connected to Linux usernames; don't get confused between them.) You can set some other configuration options that appear on the next screen, as shown Figure 8-5.

The most interesting piece here is the custom identifier for the frontend. If the frontend is using DHCP to get its network configuration, then it's possible that the IP address of the frontend is going to change. MythTV by default saves the configuration settings for a frontend based on its IP, but this doesn't work if the IP address of the machine is going to change. In that case, you're better off setting a text description of the machine, as we have done here.

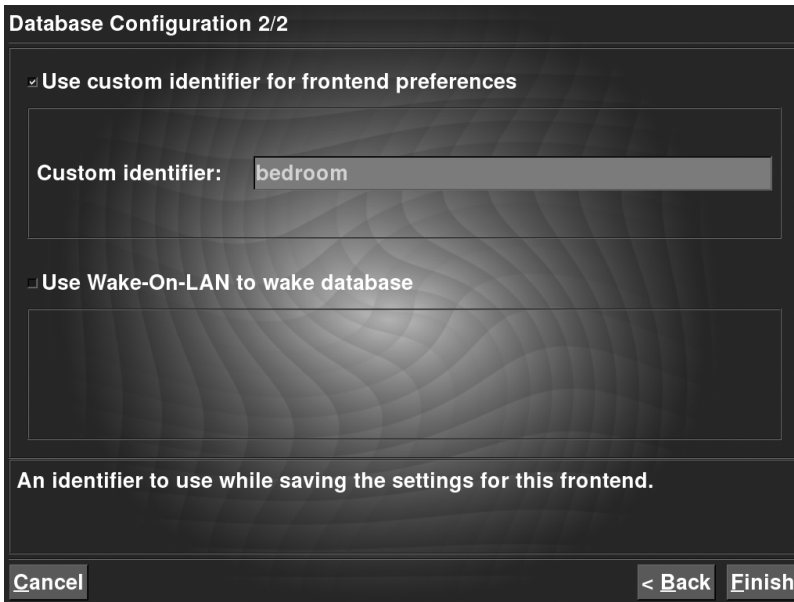


Figure 8-5. *Further configuration for the database*

When you're finished, then you should be able to run the frontend using `mythfrontend`. The theme won't have been configured yet (configuring themes is covered in the previous chapter of this book), so you'll end up with the default, which is shown in Figure 8-6.



Figure 8-6. *The new frontend running with the default theme*

Now, your new Linux frontend should be ready to go. We won't discuss here how to get DVD playback working for full-sized Linux frontend machines (although we do discuss it for Xbox Linux frontends in the next section), because that is the focus of Chapter 12.

Running a Frontend on an Xbox

We built a Microsoft Xbox frontend for a MythTV setup for a few reasons: we already had an Xbox lying around unused; the Xbox hardware is very cheap; there is plenty of remote control hardware available for the Xbox that is simple to install; and an Xbox can natively display on a regular TV set without any additional hardware, which worked well for the room to which we wanted to add a frontend. In this chapter, we're referring to Microsoft's original Xbox hardware, not the newer Xbox 360, which has not yet been hacked to run Linux. We used a Microsoft DVD remote control set for the remote control on this frontend. Building an Xbox-based frontend is really a special case of building any other Linux frontend.

The first step is to install a Linux distribution on your Xbox. This isn't as hard as it sounds, because the Xbox Linux project at http://www.xbox-linux.org/wiki/Main_Page has already solved this problem. We chose Xebian, because it's close to the Debian and Ubuntu distributions, which we use on our other systems. You can find Xebian at <http://www.xbox-linux.org/wiki/Xebian>. For Xebian to run on our version of Xbox, we needed to install a Cromwell BIOS, which means that the Xbox can no longer play Xbox games. That's not such a big deal given that we don't play computer games. There are other options here, and if you want your Xbox to remain as a game-playing machine as well, then you should read up on your options at the Xbox Linux site.

If you don't want to go the Xebian route, then you might be interested in the Xbox Media Center (XBMC) project, which also allows you to have an Xbox-based MythTV frontend. You can find more details of this at <http://sourceforge.net/projects/xbmcmythtv/>, so we won't discuss it anymore here.

We won't provide a complete Xbox Linux tutorial here, because there are plenty of those on the Internet already, including the Xbox Linux project and Xebian sites that we have mentioned already. We recommend that you walk through the Xbox Linux installation details on the Xbox Linux site at <http://www.xbox-linux.org>. They can even direct you to volunteers who will help you with the Linux install if you need that.

One note before we get started discussing the Xbox installation. We built an Xbox frontend because we already had a modified Xbox lying around. If we hadn't owned the hardware, then we would have seriously considered using an old laptop or other machine instead. The main reason for our hesitation to rush out and purchase an Xbox is that the machines are quite loud when running, and that isn't great in a room where you want to watch TV. On the other hand, the Xbox hardware is cheap and does work nicely with standard-definition televisions.

The details on how to install Xebian are at http://www.xbox-linux.org/wiki/Xebian_HOWTO#Finding_and_downloading_the_latest_version and are fairly easy to follow. Remember, when you configure your Xbox frontend, you need to set up the machine with either DHCP or an IP address that is going to work on your network. You can find a more detailed discussion about this in the "Network Configuration" section.

Note The DVD player in the Xbox is designed to play only commercially produced DVDs. This has the rather nasty side effect that not all burned CD-ROMs or DVDs work with the Xbox DVD drive. This can cause you problems when you try to install Xebian or any other Xbox Linux distribution. See the Xebian description of the problem at http://www.xbox-linux.org/wiki/Xebian_HOWTO#Media_matters for more details.

Once you have Xebian installed, which took us about 30 minutes not including burning the CD-ROM and modifying the Xbox hardware, you are ready to follow the setup instructions for MythTV's frontend. Because of the slow CPU on the Xbox, you're best off installing from packages instead of building the whole thing from source yourself (source is an option if you really want, though). To install the MythTV packages, just follow the instructions on the Xbox Linux site at http://www.xbox-linux.org/wiki/MythTV_on_Xebian_HOWTO#Install_MythTV. To briefly summarize those instructions, do something like the following:

```
$ apt-get update
$ apt-get install mythtv-backend
```

This will leave you with a mythfrontend that should just work, including remote control support for the Microsoft Xbox DVD playback-style remote controls. If you don't have a Microsoft Xbox DVD-compatible remote control, then you might need to modify the `lirc` configuration for the system in order to get the remote control to work. In that case, your best bet is to do a Google search for an existing `lirc` configuration file before creating your own.

You'll need to configure the database on the backend machine to allow the Xbox to connect to the database before anything will work. We discussed this in the "MySQL Configuration" section earlier in this chapter, along with the need to change your MySQL and MythTV backend configurations slightly from the defaults.

Once the install is finished and a database account is created, then you'll be able to run mythfrontend as the user "live" and see whether it works. ("live" is the name of the default account on a Xebian system.)

We found we had trouble with error messages about old MySQL clients:

```
xebian:~# mysql -h backendip -u mythtv -p mythconverg
Enter password:
ERROR 1251: Client does not support authentication protocol requested by
server; consider upgrading MySQL client
```

In this example, `backendip` is either the DNS name of the machine running the MythTV backend or the IP address of that machine. It's quite possible by the time you read this, though, that the Xebian-packaged version of the MySQL client will have been upgraded and that you won't see this message at all. If you experience this symptom, then you can fix the problem by using an old-style password for that one client by executing this SQL on your MySQL server:

```
update user set password = old_password('newpassword') where host = 'remoteip';
```

where `remoteip` is the IP address of that client machine, `newpassword` is the password the client should present to the server, and `old_password` is the name of a function provided by MySQL, which converts it appropriately. You can find more details of the problem and the other possible solutions at the MySQL website at <http://dev.mysql.com/doc/refman/4.1/en/old-client.html>.

You can test that the frontend works by starting it up and seeing whether there are any recordings in Media Library ► Watch Recordings. Finally, after you've done all this, you just need to get DVD playback working. To do this, you need to install the DVD plug-in. Install some other plug-ins while you're there:

```
$ apt-get install mythdvd mythgallery mythgame mythmusic mythvideo
```

This will add DVD playback support, a picture display application, arcade game simulation, MP3 playback, and playback of other video files (such as MPEG files and AVIs). Note that many of these modules expect to get their content from the file system. For example, the MythMusic plug-in expects to find the MP3s to play somewhere on the local file system. This means that if you want to have that sort of content stored centrally (perhaps on the same machine as the MythTV backend), then you'll need to export those file systems that contain content using a mechanism such as NFS—MythTV frontends retrieve their recorded programs via a built-in networking protocol, but plug-ins need to make their own arrangements (or, more properly, you do). You can read more about these plug-ins in Chapter 9.

You also need to make sure the `/dev/dvd` device file exists. If it doesn't, then just make a symbolic link to the `/dev/cdrom` file like this:

```
$ ln -s /dev/cdrom /dev/dvd
```

DVD playback actually works better than the experiences we had originally with a full-blown PC. The main reasons for this are that the remote control configuration as shipped by the MythDVD package and Xebian is much better tailored, volume control is not something we do with our Xbox (it's done by external hardware), and there is a physical eject switch on the Xbox's DVD drive that always ejects the disc, regardless of what the Xbox is currently doing.

Tip Remember that in order to play region-encoded DVDs or encrypted DVDs, you need to install the `libdvdcss` library separately from `xine` itself. They are packaged separately because of possible legal implications with `libdvdcss`. Please make sure it's legal in your country first. If it is legal, then installation is as easy as this:

```
$ sudo apt-get install libdvdcss2 libdvdcss2-dev
```

Finally, the history of how the Microsoft Xbox was hacked to run Linux is a story we find quite interesting; you can find an excellent introduction to Microsoft's Xbox hardware and how Linux came to be run on it in *Hacking the Xbox: An Introduction to Reverse Engineering* by Andrew Huang (O'Reilly, 2003).

Building a Macintosh Frontend

Setting up a Macintosh frontend is much like the other frontends we have discussed. The first step is to download a precompiled version of `mythfrontend`. We got ours from <http://collectivity.goof.com/>; specifically, we used <http://collectivity.goof.com/articles/2006/09/27/mythfrontend-0-20-for-all-mac-os-x-variants>. Download that file onto your Macintosh, and then double-click it to extract it (if you used Safari, then you can do that in the Downloads window). You can see an example in Figure 8-7. You'll end up with a disk image that Finder will mount for you. This can take quite a while, because the disk image is quite large.

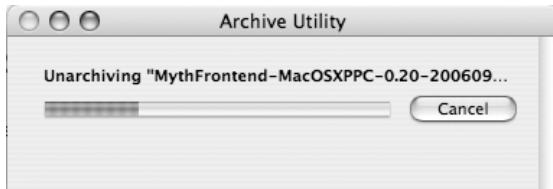


Figure 8-7. Extracting the disk image

Once you've extracted the disk image, you can copy the `mythfrontend` application that is inside it to wherever you want on your machine. We put it in the Applications folder and then created a shortcut on the Dock for it. When you start the application, you'll get the same configuration questions that you do for the other frontends. First pick a language (see Figure 8-8).

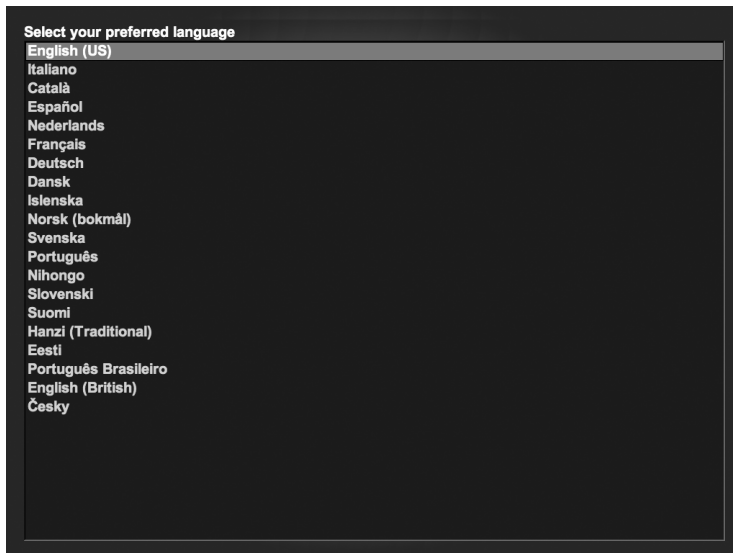


Figure 8-8. Selecting a language

Then, enter the configuration information for your backend (see Figure 8-9).

Database Configuration 1/2

Myth could not connect to the database. Please verify your database settings below.

Host name: 192.168.1.99

Database: mythconverg

User: mythtv

Password: mythtv

Database type: MySQL

The database implementation used for your server.

Cancel < Back Next >

Figure 8-9. *Configuring the backend details*

Here you need to set up the IP address (or DNS name) of the machine running the backend, the name of the MySQL user who is going to connect from this frontend, and the MySQL password that is valid for that username, connecting from the IP address that the frontend is configured with. You can set some other configuration options that appear on the next screen, as shown in Figure 8-10.

The most interesting piece here is the custom identifier for the frontend. If the frontend is using DHCP to get its network configuration, then it's possible that the IP address of the frontend is going to change. MythTV by default saves the configuration settings for a frontend based on its IP, but this doesn't work if the IP address of the machine is going to change. In that case, you're better off setting a text description of the machine, as we have done here.

When you're finished, you'll end up in the MythTV user interface, with the default theme. (Chapter 7 covered themes if you want more details on those.) Depending on the version of the mythfrontend you downloaded, you should also find that the Apple remote control just works without needing extra configuration.



Figure 8-10. *Further configuration for the database*

Windows

There is a Windows version of mythfrontend as well, with its home page located at <http://winmyth.sourceforge.net/>. However, it is early in its development and not up to the standard of the other mythfrontend options at the moment. We'll therefore not cover how to install it here and will instead recommend that you pursue some of the other viewing options, discussed next, if you want to watch your recordings on Windows machines.

Other Remote Viewing Options

You have some options for viewing recordings from MythTV remotely that don't include setting up a frontend, and the following sections are devoted to these options. These options should be attractive to you if setting up a frontend is too hard (for example you want to watch the videos on a Windows machine) or if there is a low-bandwidth connection between the backend and the machine you want to watch the videos on (for example you want to watch videos over the Internet).

Transcode and Download

Transcoding your MythTV recordings will have two effects: it will make the files smaller (which can be a big benefit if your network link is slow), and it converts the video to formats that are better supported by generic video players such as those you probably already have installed. Chapter 5 covered transcoding videos, so we won't cover that again here. It is worth saying that the divx format that nuvexport can export to is probably the best supported video format for various operating systems, so it is a good first choice when you are experimenting with what

your playback machine supports. You can also set up `nuvexport` to transcode videos automatically, which was also covered in Chapter 5.

If you would like to make the transcoded videos available on the Internet, then the easiest way is to have Apache serve the contents of the directory that your transcoded files are saved to and then access them remotely via HTTP. If you want to make the videos available on your LAN only, then you might consider NFS or SMB depending on your operating system choice.

Use a Player That Understands MythTV Formats

The drawback to transcoding video is that it is expensive in CPU time, which means you might find you have a hard time keeping up with the content you are recording if you are recording a lot. In that case, you might just want to copy the files in the MythTV recording format and then find a player that supports playing back the MythTV video format. The catch with this, of course, is that the files will be bigger.

One problem with this is that the MythTV NUV format is actually a container format that can include video compressed with a number of different codecs. MPlayer is a good choice of cross-platform video player that supports NUV. You can download copies of MPlayer for Linux, Windows, and Macintosh at <http://www.mplayerhq.hu/design7/dload.html>. Installing MPlayer is well documented at <http://www.mplayerhq.hu/DOCS/HTML/en/install.html>.

Another catch is that the filenames for the MythTV recordings are hard to understand, because they are not based on the show titles. For example, here is a subset of the listings in one of our video directories at the moment:

```
$ ls | head -10
1002_20060730203000.mpg
1002_20060730203000.mpg.png
1002_20060806203000.mpg
1002_20060806203000.mpg.png
1005_20060729180000.mpg
1005_20060729180000.mpg.png
1005_20060730180000.mpg
1005_20060730180000.mpg.png
1005_20060801173000.mpg
1005_20060801173000.mpg.png
```

The `.png` files here are thumbnails to be displayed in various user interfaces, and the `.mpg` files are MPEG-2 video from one of our Hauppauge PVR 350 cards (which records natively in MPEG-2). These filenames can make it hard to determine which video file to play with MPlayer. There are three options to determine the filenames that map to the shows you want to watch. We'll cover them in the following sections.

Using `nuvexport` to Determine Filenames

`nuvexport` can tell you the filename of videos. Chapter 5 covered `nuvexport` much more completely (including how to install it and the most useful command-line options), but here is a simple example of how to determine the filename of a video based on a search:


```
$ nuvexport --title "Whose Line Is It Anyway?" --search-only
```

```
Loading MythTV recording info.
```

```
99%
```

```
Matching Shows:
```

```
    title: Whose Line Is It Anyway?
  subtitle: Untitled
   chanid: 1052
  starts: 20060724223000
   ends: 20060724230000
 filename: /var/video//var/video/1052_20060724223000.mpg
```

```
    title: Whose Line Is It Anyway?
  subtitle: Untitled
   chanid: 1052
  starts: 20060725223000
   ends: 20060725230000
 filename: /var/video//var/video/1052_20060725223000.mpg
```

The `--search-only` flag tells `nuvexport` to not transcode the video and just output search results instead. You can see here that we have two episodes of *Whose Line Is It Anyway?* recorded, and their filenames are in the output next to the filename label.

Using MythWeb to Determine Filenames

MythWeb, which we discuss in Chapter 11, gives you a graphical way of finding recorded shows, and if you hover over the thumbnail of the video, then the filename of the recording is included in the URL. In fact, clicking the thumbnail will download the video and either save it on disk or start playing it in a video player depending on your browser preferences. See Chapter 11 for more details of MythWeb.

Using User Jobs to Determine Filenames

You can also create a MythTV user job that creates a symlink from a descriptive name for a show to the MythTV recording file. Chapter 5 includes complete coverage of user jobs, but, briefly, a user job like the following one will create a symlink between the real video file and a file with a descriptive name in `/data/links/`:

```
$ ln -s %DIR%/FILE% /data/links/`echo %TITLE%-%SUBTITLE% | sed 's/ \/_/g'`
```

The `sed` portion of this command line is needed to turn spaces and other illegal filename characters in the destination filename into underscores.

MythWeb

Finally, for these other options, MythWeb, which is discussed in Chapter 11, offers a download link for videos. Click the thumbnail next to a video, and your browser will either download or start playing the video, depending on your browser settings.

Conclusion

In this chapter, we covered how to set up frontends on machines other than the one running your MythTV backend. We also talked about the generic setup steps and then how to install frontends on Linux, on Xbox machines running Linux, and on Macintosh machines. Finally, we talked about other options if you don't want to go the whole way to installing a frontend.

The next chapter covers all the plug-in functionality available in MythTV, which lets you perform tasks such as check the weather, manage and play your MP3s and videos, and do cool stuff like look after your Netflix queue. Read on for details about that and much more.