Now we have set up the backup system.

6. To view all backup versions:

   ```
   $ git log
   ```

7. To revert back to any previous state or version, look into the commit ID, which is a 32-character hex string. Use the commit ID with `git checkout`.

   For commit ID 3131f9661ec1739f72c213ec5769bc0abefa85a9 it will be:

   ```
   $ git checkout 3131f9661ec1739f72c213ec5769bc0abefa85a9
   ```

   To make this revert permanent:

   ```
   $ git commit -am "Restore @ $(date) commit ID:
   3131f9661ec1739f72c213ec5769bc0abefa85a9"
   ```

   In order to view the details about versions again, use:

   ```
   $ git log
   ```

8. If the working directory is broken due to some issues, we need to fix the directory with the backup at the remote location. We can recreate the contents from the backup at the remote location as follows:

   ```
   $ git clone user@remotehost:/home/backups/backup.git
   ```

It will create a directory backup with all contents.

While `git` is pretty good for keeping versioned copies of text files which includes documents, source code and so on, it's not a good idea to use `git` for a large amount of binary-only data. For example, it's not sensible to use `git` to backup/version control your photo collection. The reason for this is `git` keeps whole files instead of differences when it comes to binary files, and this will make it occupy a huge amount of space.

Creating entire disk images using fsarchiver

`fsarchiver` is a tool which can save the contents of a complete `filesystem` to a compressed archive file. Due to these abilities, it is one of the most complete and easy to use tools for backup.

`fsarchiver` is the successor of `partimage` - the well-known **filesystem** backup solution. `fsarchiver` has the advantage of supporting newer **filesystems** like **ext4** when compared to `partimage`, however the latter has a minimal GUI that makes it somewhat easier to use.

Getting ready

`fsarchiver` is not installed in most of the distros by default. You will have to manually install it using your package manager. If you want more information, go to `http://www.fsarchiver.org/Installation`

How to do it...

1. Creating a backup of a filesystem/partition

   Use the `savefs` option of `fsarchiver` like this:

   ```
   fsarchiver savefs backup.fsa /dev/sda1
   ```

   where backup.fsa is the final backup file and /dev/sda1 is the partition to backup

2. Backup more than one partition at the same time

   Use the `savefs` option as earlier and pass the partitions as the last parameters to `fsarchiver`:

   ```
   fsarchiver savefs backup.fsa /dev/sda1 /dev/sda2
   ```

3. Restore a partition from a backup archive

   Use the `restfs` option of `fsarchiver` like this:

   ```
   fsarchiver restfs backup.fsa id=0,dest=/dev/sda1
   ```

   `id=0` denotes that we want to pick the first partition from the archive to the partition specified as `dest=/dev/sda1`

4. Restore multiple partitions from a backup archive

As earlier, use the `restfs` option as follows:

```
fsarchiver restfs backup.fsa id=0,dest=/dev/sda1 id=1,dest=/dev/sdb1
```

Here, we use two sets of the `id,dest` parameter to tell `fsarchiver` to restore the first two partitions from the backup to two physical partitions.

How it works...

Very similar to the way tar works, `fsarchiver` goes through the filesystem to create a list of files and then saves them to a compressed archive file. The advantage here is that unlike tar which only saves information about the files, `fsarchiver` performs a backup of the filesystem as well. This means that it is easier to restore the backup on a fresh system as it is not necessary to recreate the filesystem.