dpkg

help.ubuntu.com/12.04/serverguide/dpkg.html

dpkg is a package manager for Debian-based systems. It can install, remove, and build packages, but unlike other package management systems, it cannot automatically download and install packages or their dependencies. This section covers using dpkg to manage locally installed packages:

- To list all packages installed on the system, from a terminal prompt type:

```
dpkg -
l
```

- Depending on the amount of packages on your system, this can generate a large amount of output. Pipe the output through grep to see if a specific package is installed:

```
dpkg -l | grep
apache2
```

  Replace apache2 with any package name, part of a package name, or other regular expression.

- To list the files installed by a package, in this case the ufw package, enter:

```
dpkg -L
ufw
```

- If you are not sure which package installed a file, dpkg -S may be able to tell you. For example:

```
dpkg -S /etc/host.conf
base-files:
/etc/host.conf
```

  The output shows that the /etc/host.conf belongs to the base-files package.

  Many files are automatically generated during the package install process, and even though they are on the filesystem, dpkg -S may not know which package they belong to.

- You can install a local .deb file by entering:

```
sudo dpkg -i zip_3.0-
4_i386.deb
```

  Change zip_3.0-4_i386.deb to the actual file name of the local .deb file you wish to install.

- Uninstalling a package can be accomplished by:

```
sudo dpkg -r
zip
```

Uninstalling packages using dpkg, in most cases, is NOT recommended. It is better to use a package manager that handles dependencies to ensure that the system is in a consistent state. For example using dpkg -r zip will remove the zip package, but any packages that depend on it will still be installed and may no longer function correctly.

For more dpkg options see the man page: man dpkg.