

# CONTRIBUTED DOCUMENTATION

This document is NOT supported by Computing Services.  
DO NOT contact the Help Center with questions on this document.

## Using the Andrew Unix Package

---

### Overview

Package is a set of programs and configuration files that can be used to:

- move files from one location to another
- confirm the existence and permissions of files
- ensure that multiple copies of a file are identical

Package is an integral part of the Andrew Unix environment. Every Andrew machine runs Package on startup to ensure that all system files are present and up to date. Within the package configuration file (`/etc/package.proto`), it is possible to change permissions, add services, and even specify which version of the kernel runs on the machine.

It should be noted that there is a similar tool called Depot that also runs on Andrew unix machines. Depot will be discussed more thoroughly elsewhere, but for simplicity, consider Package the program that operates on fundamental system files, while Depot operates on programs and packages that are directly run by the end user.

### Syntax

Package configuration files consist of commands, one per line, with the end of line functioning as the command terminator. Whitespace (tabs or spaces) separate items, and the amount of whitespace is not significant. In general, program control lines begin with a percent sign, while file commands begin with a letter of the alphabet. Comments begin with an octothorpe (pound sign), and lines containing only whitespace are ignored.

#### Variables

Package uses variables extensively. Variables are defined with the `%define` command. A line should contain a percent sign (to designate a program control line), the define command, the name of the variable that's being assigned a value, and the value that's being assigned to the variable.

```
Example:  
% define variablename value  
% define numberofbits 32
```

Then, to use the variable (in an `ifdef` statement, for example, which is explained below), you wrap the variable in braces and precede the braces with a dollar sign.

```
Example:  
% ifdef ${variablename}
```

If, for some reason, it's no longer desirable for a variable to have a value, it can be undefined with the undefine command. This is simply a percent sign, the undef command, and the variable to be undefined.

*Example:*

```
% undef variablename
```

### Including files with %include

Package files can include other package files at any point. When a file is included, it is included "inline", not as a subordinate. All definitions available to the calling configuration file are available to the included file, and anything defined in the included file is available to the parent. Include statements should use absolute pathnames and they may include variables.

*Example:*

```
% include /local/extra/package.extras
% define cell andrew.cmu.edu
% include /afs/${cell}/somewhere/or/another/package.config
```

### Flow control: ifdef, ifndef, else, endif

Package provides conditional interpretation of configuration files via the ifdef and ifndef commands. The ifdef and ifndef commands are similar to the if, then, and else commands of many programming languages.

When an ifdef statement is encountered, Package checks to see whether the variable name after it is defined. If the variable is defined, the lines below the ifdef are read and parsed until the appropriate else or endif is encountered. If the variable is not defined, the lines below the endif are not parsed until an appropriate else or endif is encountered.

An ifndef is effectively "if not defined". It functions like the ifdef statement, except that functions below it are only parsed if the variable is not defined.

When an else is encountered, the truth of the preceding ifdef is reversed. Lines below the else are parsed if the ifdef or ifndef was false, while the lines are ignored if the ifdef or ifndef was true.

All ifdefs MUST be concluded with an endif command. Ifdefs and ifndefs may be nested, and frequently are.

*Example:*

```
% define debug 1
% ifdef debug
% ifndef release
% include /local/package/package.debug
% endif
% else
% include /local/package/package.release
% endif
```

### File Commands

This is the heart of package's purpose. All of the above commands are used to select the file commands that are performed. The file commands ensure that a file exists and, in most cases, that it is identical to a "master" file located elsewhere.

### Generic Format

Each of the file command lines begins with a letter (not a percent sign) indicating the type of file that is being operated on. It is optionally followed immediately (without whitespace) by one or more options. Then, after whitespace, the filename followed by more specific options for each type of device. Often, the owner, group, and octal file permissions.

## B: Block Special Devices

Block special devices are designated with a B in the package configuration file. This is followed by the filename, the major number, and the minor number, followed optionally by the owner username, the groupname, and the permissions in octal.

*Example:*

```
B /dev/fd0 2 0 root wheel 600  
No options are currently used
```

## C: Character Special Devices

Character special devices are designated with a C in the package configuration file. Otherwise, its format is very similar to the block special character. The C is followed by the filename, the major number, and the minor number, followed optionally by the owner username, the groupname, and the permissions in octal.

*Example:*

```
C /dev/audio 14 4 root wheel 600  
No options are currently used
```

## D: Directories

Directories are designated with a D in the package configuration files. The D command can take several options. These options are:

- DA Directory with an absolute path
- DQ After updating this directory, update with status 4. This will cause a reboot on Andrew systems.
- DR Unconfigured files in the directory will be removed
- DT copy ownership and permissions from the prototype directory DU Directory. Update may not be inhibited by file mode.
- DW If there's a conflict between the existing directory and the proto directory, the proto directory supercedes the existing one.
- DX lost+found directory

These options may be combined so long as the options do not conflict.

After the D and any options, the directory is listed followed by the "master" if A is selected, then optionally the owner username, the owning group, and permissions in octal.

*Examples:*

```
DR /usr/local/depot ${treemode}
```

D /stuff root wheel 1777

## F: Regular Files

Files are designated with an F in the package configuration files. The F command can take several options. These options are:

- FA File followed by absolute path to "master" file.
- FI (Initialize) Only copy this file from the master location if no file is currently present.
- FO (save Old) When this file is replaced by the master file, save the old file as .old.filename.
- FQ After updating file, exit with status 4. This will cause a reboot on Andrew systems.
- FU Normally, when a file is not writeable by its owner, package will not update or remove the file. FU allows package to update or remove the file regardless of permissions.
- FW Update wins if conflict exists.

These options may be combined as long as the options do not conflict.

After the F and any options, the directory is listed followed by the "master" if A is selected, then optionally the owner username, the owning group, and permissions in octal.

*Examples:*

```
FOAQ /lib/modules/${linux24ver}/fs/afs.o ${afs}/root.client/usr/vice/etc/modload/ libafs.o.  
${linuxspec}-${linux24-afsver} ${textmode}
```

```
FOAQ /lib/modules/${linux24ver}/fs/afs.o ${afs}/root.client/usr/vice/etc/modload/ libafs.o.  
${speciallibafsmachine}.${linuxspec}-${linux24-afsver} ${textmode}
```

```
FIA /etc/snmpd.agentinfo ${wsadmin}/lib/null ${textmode}
```

```
FIA /var/log/pacct ${wsadmin}/lib/null root wheel 644
```

## L: Symbolic Links

L

Symbolic Links are designated with an L in the package configuration files. The L command can take several options. These options are:

- LA Link followed by absolute path to "master" file
- LQ After updating link, exit with status 4. This will cause a reboot on Andrew systems.

These options may be combined so long as the options do not conflict.

After the L and any options, the link is listed followed by the "master" file that is being linked to.

*Examples:*

```
L /usr/lib/ppd ${host}/printcap ${treemode}  
LA /etc/domain /usr/domain
```

## Pipes

P

Pipes are designated with a P in the package configuration files. The P command can take one option. This option is:

- PQ After updating the pipe, exit with status 4. This will cause a reboot on Andrew systems.

These options may be combined so long as the options do not conflict. After the P and any options, the pipe is listed, then optionally the owner username, the owning group, and permissions in octal.

*Example:*

```
P /dev/initctl root wheel 600
```

## Sockets

Sockets are designated with an S in the package configuration files. The S command can take one option. This option is:

- SQ After updating socket, exit with status 4. This will cause a reboot on Andrew systems.

After the S and any options, the socket is listed, then optionally the owner username, the owning group, and permissions in octal.

*Example:*

```
S /dev/log root daemon 666
```

## Other

Anything else is designated with an N in the package configuration files. The N command can be used to mark anything. It may be used in places where there may be either a file or a directory with the same name in a location, for example.

The N command can take one option. This option is:

- NQ After updating, exit with status 4. This will cause a reboot on Andrew systems.

After the N and any options, the location of the item is listed, then optionally the owner username, the owning group, and permissions in octal.

The N option is very infrequently used.

---

## Frequently Asked Questions

- [How do I know if Package completed successfully?](#)
- [How do I create a local root password?](#)
- [I'd like to use SSH on my machine. How can I do that?](#)

- [I'd like to allow other people to use the machine when someone's on the console. How do I do that?](#)
- [I'd like to run a web server on my machine. What's the easiest way to set one up?](#)
- [How can I improve AFS performance?](#)
- [How do I restrict logins on my machine so that only people in my department can log in?](#)
- [How do I add a local user to the machine?](#)
- [I want to install some software in Sun's package format. How can I do so under Andrew Solaris?](#)
- [I have a small root partition and a large /usr partition. How can I move /tmp to my /usr partition?](#)
- [I'd like to offer files via anonymous FTP. How can I do so?](#)
- [I'd like to offer files via NFS. How can I do so?](#)

## Answers

### How do I know if Package completed successfully?

Look at the end of the /var/log/package file. It should end with "sync", "done", then the date when package completed. Also, if package failed, you can look at /afs/andrew/wsadmin/workstations/bad/@sys/`hostname`

This file exists in AFS, so it can be seen from any machine if you use the appropriate system type instead of @sys and the hostname of the machine where package failed instead of **hostname**.

[Return to questions](#)

### How do I create a local root password?

The following steps assume that you don't have a root instance. If you do, it's actually a simpler process than shown below (you can skip step 2), but you won't need a local root password in that case.

**Note:** The following commands should first be added to the /etc/package.proto file.

1. Create an encrypted version of the root password with the rootpass program. Store the result somewhere on the root partition. Use a command like this:  
**/usr/local/etc/rootpass > /tmp/passwd.change**
2. Boot the machine into single user mode. To do this, at the OK prompt on a sun, type **boot -s**. On a linux machine, type **linux init=/bin/sh** at the LILO prompt. On a linux machine, you'll also need to remount the disk in read/write mode using **/sbin/fsck ; mount -n -o remount /**
3. Copy the file you created in step 1 to **/etc/passwd.change** (cp /tmp/passwd.change /etc/passwd.change)
4. Add these lines to /etc/package.proto:  
**F /etc/passwd.change**  
**%define haspasswd.change**
5. Reboot.

[Return to questions](#)

### I'd like to use SSH on my machine. How can I do that?

Use the command **%define doessshd**

**Note:** This command should first be added to the `/etc/package.proto` file.

[Return to questions](#)

**I'd like to allow other people to use the machine when someone's on the console. How do I do that?**

```
%define netopen
```

**Note:** This command should first be added to the `/etc/package.proto` file.

[Return to questions](#)

**I'd like to run a web server on my machine. What's the easiest way to set one up?**

```
%define doesapache
```

**Note:** This command should first be added to the `/etc/package.proto` file.

This will place an apache installation in `/usr/www`. The document root will be `/usr/www/tree` and configuration files can be found in `/usr/www/conf`. Please take a look at `/usr/www/conf/httpd.conf`.

[Return to questions](#)

**How can I improve AFS performance?**

If you have the disk space, you can increase the size of your local AFS file cache. To do this:

```
%define specialcacheinfo 100000
```

**Note:** This command should first be added to the `/etc/package.proto` file.

The number is the size of your cache in Kilobytes. The above line creates a 100 MB cache. The specialcache trick only works up to a few hundred megabytes. For larger cache sizes, you have to modify the `/usr/vice/etc/cacheinfo` file. We do not support or recommend doing that.

[Return to questions](#)

**How do I restrict logins on my machine so that only people in my department can log in?**

The following command should first be added to the `/etc/package.proto` file. Next, you need to create the file `/etc/user.permits`. This file should list the usernames or pts groups that are allowed to log in to the machine, one per line. Then:

```
F /etc/user.permits
%define hasuser.permits
```

If you'd like to store the file somewhere else (perhaps in a departmental folder), package can copy the file to the machine on each reboot. If the file can be found in `/afs/andrew.cmu.edu/mydepartment/etc/user.permits`, you can do this:

```
FA /etc/user.permits /afs/andrew.cmu.edu/mydepartment/etc/user.permits
```

[Return to questions](#)

## How do I add a local user to the machine?

You shouldn't. There's no good provision for doing so within the Andrew Unix system. You can contact [advisor@andrew.cmu.edu](mailto:advisor@andrew.cmu.edu) to set up accounts for other people.

If you really need to set up an account IMMEDIATELY, you can perform the following procedure. We do not recommend it, we do not support it, and if you have any problems, our first suggestion will be to undo this procedure. Most of these steps should be done as root.

**Note:** This command should first be added to the `/etc/package.proto` file.

1. First, lock `/etc/passwd` to prevent any future changes to this file by the system. This means that as users are added and removed to the main Andrew system, they will not show up on your machine. To do this, add  
**F /etc/passwd to /etc/package.proto**
2. Generate a password hash for the user. You can use `/usr/local/etc/rootpass` for this purpose.
3. Add an entry for the local user to `/etc/passwd`. Be sure to choose a UID (the third field in the file) that is not in use anywhere else in the file. See a book on unix system administration ( "Unix System Administration Handbook" by Evi Nemeth et al. is the canonical recommendation) for more information on the format of this file. A line might take the form:

```
beegle:RE79B6HcSP7oM:47537:10:William C Beegle:/usr/localuser/beegle:/bin/csh
```

**Note:** The local password in the second field instead of the X that most users have, and `/usr/localuser/beegle` instead of a directory in AFS.

4. Create the home directory that you chose for the user when you added the `/etc/passwd` entry. Copy in default configuration files. Give the user ownership of of the directory. To do that, you might do the following:

```
mkdir /usr/localuser/beegle  
cp -R /usr/local/lib/proto/account/* /usr/localuser/beegle  
chown -R beegle.10 /usr/localuser/beegle
```

Then, protect your newly created directory in `/etc/package.proto` with a line like this:

```
D /usr/localuser/beegle
```

5. The local account should now be ready to use.

[Return to questions](#)

## I want to install some software in Sun's package format. How can I do so under Andrew Solaris?

We don't recommend or support doing this. However, if necessary, you can do so by adding the line

```
%define solaris_package_install_host
```

to `/etc/package.proto`, rebooting, then using Sun's package tools as usual. After installing packages, you may need to protect relevant files and directories with F and D commands in package.

**I have a small root partition and a large /usr partition. How can I move /tmp to my /usr partition?**

Adding the line

```
%define /tmplink
```

to /etc/package.proto and rebooting will turn /tmp into a symlink that points to /usr/tmp.

[Return to questions](#)

**I'd like to offer files via anonymous FTP. How can I do so?**

Add the line

```
%define doesAnonymousFTP
```

to /etc/package.proto and reboot.

[Return to questions](#)

**I'd like to offer files via NFS. How can I do so?**

This is not recommended or supported. If necessary, the process varies by operating system.

**For Andrew Solaris**, create an appropriate /etc/exports file (see **man exports** for details) and add the following lines to /etc/package.proto to protect your exports file:

```
%define hasexports
# Any %include files should go here
F /etc/exports ${textmode}
```

When the machine is rebooted, nfsd will start.

**For Andrew Linux**, you'll need to create a script at /etc/rc.d/init.d/nfs to start rpc.rquotad (if necessary), rpc.mountd, and rpc.nfsd. Once you've done that, you'll need something similar to the following to protect that script, the temporary files used by Linux NFS, and the links necessary to call your script on startup.

```
%define has/etc/rc.d/rc3.d/K20nfs
%define has/etc/rc.d/rc3.d/S60nfs
%define has/var/lib/nfs/etab
%define has/var/lib/nfs/xtab
%define has/var/lib/nfs/rmtab
# Any %include files should go here
LA /etc/rc.d/rc3.d/S60nfs /etc/rc.d/init.d/nfs ${binmode}
FI /var/lib/nfs/etab ${machine} ${textmode}
FI /var/lib/nfs/xtab ${machine} ${textmode}
FI /var/lib/nfs/rmtab ${machine} ${textmode}
```