

# FAQ

This is collection of some of the most common questions and answers likely to be asked about the software on this site. If you have other questions, please feel free to submit them to our support email address. We cannot answer questions about the operation of the software, just installation and initial startup. Please also see also our [Download and Installation Instructions](#) page.

Another excellent source of help are the pages at [Other Useful Solaris Sites](#). or the developer links for each package.

Click on the + sign to expand the question and see the answer or discussion.

---

## Site Question and Answers

### How are package downloads counted?

UNIX Packages downloads are counted when a user clicks ok on the download popup-confirmation button. Hence if the download fails it is still counted.

However, users may download the same package ie a2ps-4.14 for Solaris 10 SPARC multiple times, and it only counts as 1 download. So if a download does fail simply go back and re-download the package again without penalty.

*What does a 10 download 1 year subscription mean?*

It means you can download up to 10 packages over the next 12 months.

If you download say an earlier version of the package, a newer version later on, or the same package for a different OS like

a2ps-4.14 for Solaris 10 X86, it will count as a new download.

You can renew or upgrade your subscription at any time.

If you don't use all your download tokens within the 12 months, you lose them.

Don't forget that many packages require additional "dependencies" and they themselves have their own dependencies for example:

Apache-2.4.4 Solaris 10 SPARC:

- expat-2.0.1
- libiconv-1.14
- openldap-2.4.34
- o libtool-2.4.2
- coreutils-8.21
- libintl-3.4.0
- gmp-4.2.1
- o db-4.7.25.NC
- pcre-8.32
- sqlite-3.7.13
- o readline-6.2
- sasl-2.1.25
- zlib-1.2.7
- openssl-1.0.1e
- gcc-3.4.6

---

## Package Question and Answers

### Where can I download the Solaris Operating System?

Oracle Solaris information and availability: [Solaris 10 & 11](#)  
Previous releases of Solaris OS we believe are no longer

available, best to consult with your Oracle Sales/Account/Support contacts for official information.

UNIX Packages is not an authorized Oracle re-seller, hence we can not supply past or present copies of the Solaris OS. We have found a number of previously owned/purchased Solaris OS's on various auction websites.

## **How do I get gzip/gunzip to work?**

Some version (usually old) of gzip is available on many levels of Solaris or you can download one of our gzip packages for your system.

For example: get gzip-1.4-sol8-sparc-local for a SPARC/Solaris 8 system.

This package is NOT gzipped or compressed. So all you need to do is run pkgadd -d gzip-1.4-sol8-sparc-local as root. This puts gzip and gunzip and other files in /usr/local/bin. If you then make sure /usr/local/bin is in your PATH, you can use gzip and gunzip to create or uncompress .gz files.

If you have any problem, check the File Size in the File Info section of the package listing to make sure the file size is correct. Note, that if you download via a non-UNIX machine (like Windows), you might get a corrupted file. We can only suggest you don't move files through any non-UNIX machine on the way to your Solaris machine.

## **When download a package, it appears to download OK, but when attempting to use gunzip or pkgadd, it fails, why?**

It might say,

invalid compressed data – crc error!

or

pkgadd:ERROR:attempt to process datastream failed

```
-bad format in datastream table-of-contents
pkgadd:ERROR:could not process datastream from
```

when running `pkgadd -d` on the file.

What is wrong?

If you have `.gz` file but `gzip` says it is not in `gzip` format, it is likely that your browser downloaded the file and gunzipped it but did not take off the `.gz`.

We are finding that later versions of Google Chrome is doing this.

You can correct this problem by editing your browser's helper application for `.gz` files. If you can't do this, you might just need to do something like:

```
mv gcc-3.4.6-sol8-sparc-local.gz gcc-3.4.6-sparc-local
pkgadd -d gcc-3.4.6-sol8-sparc-local    (as root)
```

replacing the `gcc` file example with whatever file you are trying to work on.

Or try using Firefox or IE to do the download.

It is also possible that the file was downloaded and the `.gz` was removed, but the file was not gunzipped. In this case, you need to do

```
mv gcc-3.4.6-sol8-sparc-local gcc-3.4.6-sparc-local.gz
gunzip gcc-3.4.6-sol8-sparc-local.gz
pkgadd -d gcc-3.4.6-sol8-sparc-local    (as root).
```

Lastly, it is possible that a `.gz` file was downloaded, but is not the correct size. Check the `.gz` file size with the File Size given in the File Info part of the package listing. The `'file'` command run on a package will often tell you if a file is compressed or not. See the man file for details on this.

**I try to execute a program like `xftp` and get a**

## message like ld.so.1: xftp: fatal: libXm.so.3: Can't open file: errno=2 Killed. What is going wrong?

The libXm.so.3 library is in the /usr/dt/lib directory on most Solaris systems. If you do not have this library directory in your library path, you can add something like the one we use:

```
setenv LD_LIBRARY_PATH
/usr/local/lib:/usr/openwin/lib:/usr/dt/lib
```

in your .login, .cshrc, or similar file depending on the shell you are using. Then either logout and login again or do a source .login or source .cshrc or similar in your home directory. In other programs, other libraries might not be found and you will need to find them and put their directories in the LD\_LIBRARY\_PATH. It is possible that your system might not contain the library you need if you did not install it when your operating system was installed.

If you run the command like:

```
[%] bin> ldd ddd
libXm.so.4 => /usr/lib/libXm.so.4
libXp.so.1 => /usr/openwin/lib/libXp.so.1
libXaw.so.5 => /usr/lib/libXaw.so.5
libXmu.so.4 => /usr/lib/libXmu.so.4
libXext.so.0 => /usr/lib/libXext.so.0
libXt.so.4 => /usr/lib/libXt.so.4
libSM.so.6 => /usr/lib/libSM.so.6
libICE.so.6 => /usr/lib/libICE.so.6
libX11.so.4 => /usr/lib/libX11.so.4
libsocket.so.1 => /usr/lib/libsocket.so.1
libnsl.so.1 => /usr/lib/libnsl.so.1
libcurses.so.1 => /usr/lib/libcurses.so.1
liby.so.1 => /usr/lib/liby.so.1
libstdc++.so.2.10.0 =>
/usr/local/lib/libstdc++.so.2.10.0
libm.so.1 => /usr/lib/libm.so.1
libc.so.1 => /usr/lib/libc.so.1
```

```
libdl.so.1 =>    /usr/lib/libdl.so.1
libmp.so.2 =>    /usr/lib/libmp.so.2
/usr/platform/SUNW,Ultra-5_10/lib/libc_psr.so.1
```

You can see what libraries an executable like ddd expects to find.

If you do not have these installed, you may get errors when starting a program. In Solaris 8 or higher, if you want to add a directory to the library search path, you can use the crle command. Do a man crle for more details.

**I have installed gcc, but when I try to compile something I get: gcc: installation problem, cannot exec 'as' :No such file or directory.**

The 'as' program can be found in the directory  
/usr/ccs/bin/

You will need to add this to your UNIX PATH. If you cannot find 'as' in the above directory, you may not have installed all of the Sun packages. See the next question also.

Note that some programs want to use the GNU version of 'as' and not the Sun version. You can get this version in the binutils package which puts GNU 'a' in /usr/local/bin.

**When I try to compile a program using gcc, I get a message like: /usr/ccs/bin/ld: cannot open values-Xa.o: No such file or directory**

Here is a response to this from the Solaris FAQ at <http://www.wins.uva.nl/pub/solaris/solaris2/>:

6.2) What about the linker, the assembler and make?

Solaris ships with everything you need, with the exception of the compiler itself. Solaris ships with the include files, make, an assembler, linker and libraries. All this stuff lives in /usr/ccs/bin (add it to your PATH before /usr/ucb if that's

in your PATH as well) /usr/ccs/lib and /usr/include.  
If you still can't find it, make sure you have the following packages installed on your system:

```
for tools (sccs, lex, yacc, make, nm, truss, ld, as):
    SUNWbtool, SUNWspot, SUNWtoo
for libraries & headers:
    SUNWhea, SUNWarc, SUNWlibm, SUNWlibms,
    SUNWdfbh, SUNWcg6h, SUNWxwinc, SUNWolinc,
    SUNWxglh
for 64 bit development:
    SUNWarcx, SUNWbtoox, SUNWdplx, SUNWscpux, SUNWspox,
    SUNWtoox
for ucb compat:
    SUNWsra, SUNWsrh
```

These packages are all on the Solaris 2.x CDs, except that some packages may only exist in some releases and not in others. Some packages may be on separate CDs, such as the "Desktop/CDE" CD, but all are part of the Solaris "bundle".

**When I try to compile a test program, I get the message: In file included from t.c:1: /usr/local/lib/gcc-lib/sparc-sun-solaris2.5/2.7.2.1/include/stdio.h:17: sys/feature\_tests.h: No such file or directory. There is no sys/feature\_tests.h on my system.**

You need to install the SUNWhea package. This puts feature\_tests.h in

```
/usr/include/sys
```

. This package is on the Solaris 2.x CD.

## Where do the packages place the files once installed?

The large majority of packages install in `/usr/local` subdirectories, like `/usr/local/bin` for executables, `/usr/local/lib` for libraries, etc. Some programs like Apache 2 install in `/usr/local/apache2`.

In the package More Info listings, it will state where the files will go after running `pkgadd`. `/usr/local` is the default directory chosen by the large majority of software developers, so we have chosen to do that as well. We try to put package documentation in `/usr/local/doc` subdirectories for convenience. If you want to install programs elsewhere, you can compile your own source (which we have on our servers for all packages) or we can do that for a fee.

It is important that you consider putting `/usr/local/bin` in your environment's `PATH`, `/usr/local/lib` in your `LD_LIBRARY_PATH` (or use `crle`), `/usr/local/man` and `/usr/local/share/man` in your `MANPATH`, and so forth. Special programs like `openssl` need `/usr/local/ssl/lib` in the library path.

## How do I make Apache2 work on an NFS share?

1. install all the packages under `/usr/local`
2. Mount NFS file system for `/opt/apache2`
3. Move contents `/usr/local/apache2` to `/opt/apache2`
4. Make the following modifications in `/opt/apache2/apachectl` script (or create a new startup script)

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/apache2/lib
export LD_LIBRARY_PATH
HTTPD='/opt/apache2/bin/httpd'
```

5. Make the following mods in `httpd.conf`

```
ServerRoot "/opt/apache2"
```

```
        LoadModule      slotmem_shm_module
modules/mod_slotmem_shm.so
    LoadModule rewrite_module modules/mod_rewrite.so
    DocumentRoot "/opt/apache2/htdocs"
    <Directory "/opt/apache2/htdocs">
    <Directory "/opt/apache2/cgi-bin">
    RewriteEngine on
    RewriteCond %{REQUEST_METHOD} ^(TRACE|TRACK)
    RewriteRule .* - [F]
```

6. Startup Apache using the following command line options

```
/opt/apache2/bin/apachectl      start      -d
/opt/apache2/htdocs -f /opt/apache2/conf/httpd.conf
```

## **I want to install TeX on my system. What are the steps?**

The TeX system and installation method has been significantly changed since the packages here were made. These days, we suggest you use the TeX Live system at [TeX Live](#) where they have instructions on how to get a very recent version of TeX and related programs for Solaris and other systems. If you still want to try to install our older TeX version, email us at the support address.

## **Where can I find programs for Solaris 2.4 or earlier?**

We currently support SPARC/Solaris 2.5.1, 2.6, 7-11 and Intel/Solaris 8-11. If you have a very old system and are willing to give us access, it might be possible, for a fee, to compile programs for you.

**I was trying to download from this site and found that my virus software was going crazy every time I got about half way through the download.**

The software here consists of gzipped pkgadd files. The files contain executable programs which may appear to contain viruses to some virus scanning software on Windows machines. This is a false positive virus warning. Further, the files on this site were not meant to be downloaded to Windows machines. They should be downloaded to Solaris machines.

Windows downloads are often corrupted or in some other way mangled. If you must download to a Windows machine, you can ignore any warnings you may get. But, once you move the files to your Solaris machine, check the file size using the File Info link in the package listing. If you do not get the correct size for the file, then it has either been corrupted or uncompressed by your Windows software.

If for some reason you have virus software for Solaris, then you probably need to tell your virus detector vendor that they need to look at their software and find out why it is giving false positives.

A user who had this problem points out:

*I figured out that when I tried to open some packages they had been damaged and that the virus scanner plus using Internet Explorer was the culprit. I pulled the packages down using the command line ftp and then scanned them and there was no problem. I think that the Solaris code freaks out the virus scanner when it was being downloaded through Internet Explorer. This combination completely destroys the package rendering it useless.*

**Is this software available on a CD or DVD? Can I get software custom compiled or in .tar.gz rather than pkgadd format?**

Yes. Contact us for details on services and costs.

**I am getting an error message like: cpio: Cannot write reloc/lib/gcc-lib/sparc-sun-solaris2.8/2.8.1/cc1plus, errno 28, No space left on device. What does this mean?**

There are at least four directories that may be used in the pkgadd installation process. These are the directory where you downloaded the package and gunzipped it, the /usr/local or /opt directory where the final installation is done, the /var/tmp or /var/spool/pkg directories where pkgadd puts intermediate files.

The /var/spool/pkg directory is usually the culprit for this sort of error message. You can gain space in this directory by removing items from it, but unless you really know what you are doing, this is not a good idea. It is better to create a directory in a larger file system, move the items in /var/spool/pkg to that new directory and then make a soft link from the new directory to /var/spool/pkg (you have to delete /var/spool/pkg first of course). If you do not know what you are doing, find a system guru to help. It is always a good idea to back up anything to plan to delete or move before you do it.

If you are setting up a new machine, remember to make your /var partition larger than the default.

Another option to deal with a small /var file system that might be better for some is to use the -s option to pkgadd. Suppose we have a package program-1.00-sol8-sparc-local.gz that we have placed in /tmp. Can do the following as root:

```
# cd /tmp (or where you put the program file)
# gunzip program-1.00-sol8-sparc-local.gz
# mkdir /tmp/spool (this can be any directory in a
place with a lot of disk space)
# pkgadd -s /tmp/spool -d program-1.00-sol8-sparc-local
# cd /tmp/spool
# pkgadd -d .
```

Then answer the questions to finish the install.

---