

How to Fix Unmet Dependencies Error on Ubuntu

By Kevin Arrows • June 27, 2018 • 1 • 4 minutes read

Error **The following packages have unmet dependencies** has plagued Ubuntu users for a while now, and there are more and more sightings of the error on various versions of Ubuntu. The APT package management system is easy to use, but in some occasions, such as when you're mixing in third-party dependencies, you might get this error from **apt-get**.

This happens when you try to install something with the apt system via a terminal, and the installation fails with no obvious problem whatsoever. This issue isn't limited to any one particular program, but it might happen with quite a few of them. This is because the issue lies in the apt system, and not in the program you're installing.

There are fortunately quite a few solutions for this, some of which are easier to do, and others more difficult, but if you follow the instructions in the methods below, you will get rid of the error in no time.

Note: Before proceeding with any of the methods, it is advisable that you back up configurations files such as `/etc/apt/sources.list` and `/var/lib/dpkg/status`, so you can revert back any changes in case something goes wrong. This is done by using the following steps:

1. Press simultaneously the Alt, Ctrl and T on your keyboard to open a Terminal.
2. Type in `sudo cp /etc/apt/sources.list /etc/apt/sources.list.original` and press Enter.
3. When the previous command finishes running, type in `sudo cp /var/lib/dpkg/status /var/lib/dpkg/status.original` and press Enter again.

Method 1: Use the -f parameter

This is the easiest one to try, and only requires adding two more letters to the command that you're typing. Instead of using **sudo apt-get install PACKAGENAME**, where PACKAGENAME is the package you're trying to install with the apt system, use **sudo apt-get install -f**. The **-f** parameter will attempt to correct a system which has broken dependencies, after which you'll be able to install the package in question.

1. Open a **Terminal** by pressing **Ctrl, Alt** and **T** simultaneously on your keyboard.
2. Type in **sudo apt-get install -f** and press **Enter** to execute it.
3. Once it's done, type in **sudo dpkg --configure -a**, press **Enter** to run it, and run the **command from step 2** once more.

Method 2: Use Aptitude

Aptitude is an alternative of **apt-get** which you can use as a higher-level package manager. You can use it to try and install your package with it, instead of apt-get, but first you need to install **aptitude**.

1. Press simultaneously the **Ctrl, Alt,** and **T** keys on your keyboard to open a
2. Type in **sudo apt-get install aptitude** and press **Enter** to execute the command.
3. Type in **sudo aptitude install PACKAGENAME**, where **PACKAGENAME** is the package you're installing, and press **Enter** to execute it. This will try to install the package via aptitude instead of apt-get, which should potentially fix the unmet dependencies issue.

Method 3: Make sure that the restricted and universe repositories are enabled and try a better server

1. Press simultaneously **Alt** and **F2** on your computer, type in **software-properties-gtk** and press
2. In the **Ubuntu Software** tab, make sure that **all the repositories (main, universe, restricted, multiverse)** are enabled.
3. Click the list of servers where it says **Download from**, and choose
4. Click **Select Best Server**.
5. Press **Alt, Ctrl** and **T** simultaneously to open a Terminal, and type in **sudo apt-get update**, then press **Enter**. Once it's done running, try installing the software again.

Method 4: Clean the package database

A corrupted package database is a potential cause for unmet dependencies, as well as packages not installing properly. However, cleaning the package database can fix this, and you can do it with two commands, which I will explain below. First of all, however, press **Ctrl, Alt** and **T** to open a Terminal, and don't forget to hit **Enter** after the command in order to run it.

1. **sudo apt-get clean** will clean the local repository from all the retrieved package files (.deb). It will remove everything except the lock files from `/var/cache/apt/archives`, and `/var/cache/apt/archives/partial/`.
2. **sudo apt-get autoclean** will also clean up the retrieved files, but unlike the previous command, this one only removes packages that you can no longer download and are pretty much useless.

Method 5: Eliminate any held packages

Held packages are actually held because there are dependency problems and conflicts which **apt** can't solve. Eliminating such packages means that there won't be any such conflicts, and may consequently fix your issue.

1. Open a **Terminal** by pressing **Ctrl, Alt** and **T**
2. Type in **sudo apt-get -u dist-upgrade** and press If there are any held packages, it will show them, and it is yours to eliminate them.
3. First try running

“

***sudo apt-get -o
Debug::pkgProblemResolver=yes dist-
upgrade***

and see if it fixes the issue. If it exits with **X not upgraded** at the end, where X is the number of held packages, you will need to delete them one by one.

4. To remove a held package, use **sudo apt-get remove --dry-run PACKAGENAME** (PACKAGENAME is the package you're trying to remove). The **--dry-run** parameter makes sure you are informed of whatever happens next. When you've removed all packages, try installing the one that caused the problem in the first place, and see what happens.

Method 6: Purge/Remove/Disable PPAs

Personal Package Archives are repositories that are hosted on the Launchpad, and are used to upgrade or install packages that aren't usually available in the official repositories of Ubuntu. They're most commonly a cause of unmet dependencies, especially when they're used to upgrade an existing package from the Ubuntu repository. You can either disable, remove or purge them.

Disable means that packages installed from that PPA will no longer get updates.

1. Press simultaneously **Alt** and **F2**, and run **software-properties-gtk**.
2. From the **Other Software** tab, you will find two lines for every PPA, where one is for the source, and another for the compiled package. To disable a PPA, you should **uncheck both lines**.

Purge means that all packages in the selected PPA will be downgraded to the version in the official repositories, and will also disable the PPA. To install PPA Purge, you could use **sudo apt-get install ppa-purge**, but considering that the **apt** is broken, you should use this command in the Terminal (**Alt**, **Ctrl** and **T** simultaneously, then **Enter** to run):

```
mkdir ppa-purge && cd ppa-purge && wget http://mirror.pnl.gov/ubuntu/pool/universe/p/ppa-purge/ppa-purge_0.2.8+bzr56_all.deb && wget http://mirror.pnl.gov/ubuntu//pool/main/a/aptitude
```

/aptitude_0.6.6-1ubuntu1_i386.deb && sudo dpkg -i ./*.deb

Next, run **sudo ppa-purge ppa:someppa/ppa** in order to purge the selected PPA. However, since PPA Purge still doesn't remove a PPA, you can use the commands below to remove the PPA. Ignore the first one if your intentions don't include removing the installed package.

1. **sudo apt-get autoremove --purge PACKAGENAME**
2. **sudo add-apt-repository --remove ppa:someppa/ppa**
3. **sudo apt-get autoclean**

You should be able to install the necessary package afterwards.

Even though there are quite a few methods above, you should also know that it's always better to prevent such issues. You should keep your system up-to-date, only use trusted PPAs, and back up when everything is working properly so you can restore later. However, if you've forgotten to do these things, use the methods above to fix your issue, and use the prevention methods to make sure you don't

Read Next

- [Ubuntu 18.04 LTS Bionic Beaver Officially Available for Ubuntu, Kubuntu,...](#)
- [Windows 10 V1803: Update KB4458166 fixes issue of TLS 1.2 Dependencies](#)
- [Fix: Component 'MSCOMCTL.OCX' or one of its dependencies not correctly...](#)
- [Ubuntu 16.04.5 LTS Released on Heels of Ubuntu 18.04 LTS, Bundles All...](#)

[Comments](#)[Community](#)[Privacy Policy](#)

Avata

[Recommend](#) 1 [Tweet](#) [Share](#)[Sort by Best](#) ▾**Andres Casasola** • 5 months ago

Thank you so much.
Method 3 solved my issue.

 | • [Reply](#) • [Share](#) >**Andres Casasola** • 5 months ago

Thank you so much. Method 3 solved my problem :)

 | • [Reply](#) • [Share](#) >**Elon** • 8 months ago

Thank you very much, method 2 solved my issue

 | • [Reply](#) • [Share](#) >**Raul Belmonte de Souza** • 9 months ago

Thank you so much

 | • [Reply](#) • [Share](#) >**JYOTISMAN DAS** • 10 months ago

Try:
`sudo apt --fix-broken install`

 | • [Reply](#) • [Share](#) >**Elizandro G. Roos** • a year ago

in case nothing above works, try this:

```
sudo dpkg --purge --force-depends PACKAGES
```

which will purge only these packages and keep the dependences. This will help if you plan to reinstall the package (or a newer version) after

 | • [Reply](#) • [Share](#) >**kyeongjin nam** • a year ago

Thank you so much~

 | • [Reply](#) • [Share](#) >

