s

* [Partners](http://www.ubuntu.com/partners)
* [Support](http://www.ubuntu.com/support)
* [Community](http://www.ubuntu.com/community)
* [Ubuntu.com](http://www.ubuntu.com)

# [**Ubuntu Documentation**](https://help.ubuntu.com/)

* [Official Documentation](https://help.ubuntu.com/)
* [Community Help Wiki](https://help.ubuntu.com/community/CommunityHelpWiki)
* [Contribute](http://community.ubuntu.com/contribute/documentation/)
* [Page History](https://help.ubuntu.com/community/RootSudo?action=info)
* [Login to edit](https://help.ubuntu.com/community/RootSudo?action=login&login=1)

# [**RootSudo**](https://help.ubuntu.com/community/RootSudo?action=fullsearch&value=linkto%3A%22RootSudo%22&context=180)

|  |
| --- |
| Contents   1. [Background Information](https://help.ubuntu.com/community/RootSudo#Background_Information) 2. [Advantages and Disadvantages](https://help.ubuntu.com/community/RootSudo#Advantages_and_Disadvantages)    1. [Benefits of using sudo](https://help.ubuntu.com/community/RootSudo#Benefits_of_using_sudo)    2. [Downsides of using sudo](https://help.ubuntu.com/community/RootSudo#Downsides_of_using_sudo) 3. [Usage](https://help.ubuntu.com/community/RootSudo#Usage)    1. [sudo](https://help.ubuntu.com/community/RootSudo#sudo)    2. [Graphical sudo](https://help.ubuntu.com/community/RootSudo#Graphical_sudo)    3. [Drag & Drop sudo](https://help.ubuntu.com/community/RootSudo#Drag_.26_Drop_sudo) 4. [Users](https://help.ubuntu.com/community/RootSudo#Users)    1. [Allowing other users to run sudo](https://help.ubuntu.com/community/RootSudo#Allowing_other_users_to_run_sudo)    2. [Logging in as another user](https://help.ubuntu.com/community/RootSudo#Logging_in_as_another_user)    3. [root account](https://help.ubuntu.com/community/RootSudo#root_account)       1. [Enabling the root account](https://help.ubuntu.com/community/RootSudo#Enabling_the_root_account)       2. [Re-disabling your root account](https://help.ubuntu.com/community/RootSudo#Re-disabling_your_root_account) 5. [Other Information](https://help.ubuntu.com/community/RootSudo#Other_Information)    1. [Misconceptions](https://help.ubuntu.com/community/RootSudo#Misconceptions)    2. [Special notes on sudo and shells](https://help.ubuntu.com/community/RootSudo#Special_notes_on_sudo_and_shells) 6. [Remove Password Prompt For sudo](https://help.ubuntu.com/community/RootSudo#Remove_Password_Prompt_For_sudo) 7. [Reset sudo timeout](https://help.ubuntu.com/community/RootSudo#Reset_sudo_timeout) 8. [Other Resources](https://help.ubuntu.com/community/RootSudo#Other_Resources) |

Note: For help with configuring *sudo* privileges via its configuration file /etc/sudoers, please see [Sudoers](https://help.ubuntu.com/community/Sudoers).

# **Background Information**

In Linux (and Unix in general), there is a SuperUser named **Root**. The Windows equivalent of Root is Administrators group. The SuperUser can do anything and everything, and thus doing daily work as the SuperUser can be dangerous. You could type a command incorrectly and destroy the system. Ideally, you run as a user that has only the privileges needed for the task at hand. In some cases, this is necessarily Root, but most of the time it is a regular user.

**By default, the Root account password is locked in Ubuntu.** This means that you cannot login as Root directly or use the su command to become the Root user. However, since the Root account physically exists it is still possible to run programs with root-level privileges. This is where sudo comes in - it allows authorized users (normally "Administrative" users; for further information please refer to [AddUsersHowto](https://help.ubuntu.com/community/AddUsersHowto)) to run certain programs as Root without having to know the root password.

This means that in the [terminal](https://help.ubuntu.com/community/UsingTheTerminal) you should use sudo for commands that require root privileges; simply prepend sudo to all the commands you would normally run as Root. For more extensive usage examples, please see below. Similarly, when you run GUI programs that require root privileges (e.g. the network configuration applet), use graphical sudo and you will also be prompted for a password (more below). Just remember, when sudo asks for a password, it needs **YOUR USER password**, and not the Root account password.

**Please keep in mind, a substantial number of Ubuntu users are new to Linux.** There is a learning curve associated with any OS and many new users try to take shortcuts by enabling the root account, logging in as root, and changing ownership of system files.

Example: [Broken system via (ab)use of root by a new user](http://ubuntuforums.org/showthread.php?t=1877557)

Please note: At the time of the post, this was the users first post on the Ubuntu forums. While some may say this is a "learning experience", learning by breaking your system is frustrating and can result in data loss.

When giving advice on the Ubuntu Forums and IRC, please take the time to teach "the basics" such as ownership, permissions, and how to use sudo / gksu / kdesudo in such a way that new users do not break systems.

# **Advantages and Disadvantages**

## Benefits of using sudo

Some benefits of leaving **Root** logins disabled by default include the following:

* The Ubuntu installer has fewer questions to ask.
* Users don't have to remember an extra password (i.e. the root password), which they are likely to forget (or write down so anyone can crack into their account easily).
* It avoids the "I can do *anything*" interactive login by default (e.g. the tendency by users to login as an "Administrator" user in Microsoft Windows systems), you will be prompted for a password before major changes can happen, which should make you think about the consequences of what you are doing.
* sudo adds a log entry of the command(s) run (in /var/log/auth.log). If you mess up, you can always go back and see what commands were run. It is also nice for auditing.
* Every cracker trying to *brute-force* their way into your box will know it has an account named **Root** and will try that first. What they don't know is what the usernames of your other users are. Since the Root account password is locked, this attack becomes essentially meaningless, since there is no password to crack or guess in the first place.
* Allows easy transfer for admin rights, in a short term or long term period, by adding and removing users from groups, while not compromising the **Root** account.
* sudo can be setup with a much more fine-grained security policy.
* The Root account password does not need to be shared with everybody who needs to perform some type of administrative task(s) on the system (see the previous bullet).
* The authentication automatically expires after a short time (which can be set to as little as desired or 0); so if you walk away from the terminal after running commands as Root using sudo, you will not be leaving a Root terminal open indefinitely.

## Downsides of using sudo

Although for desktops the benefits of using sudo are great, there are possible issues which need to be noted:

* Redirecting the output of commands run with sudo requires a different approach. For instance consider sudo ls > /root/somefile will not work since it is the shell that tries to write to that file. You can use ls | sudo tee -a /root/somefile to append, or ls | sudo tee /root/somefile to overwrite contents. You could also pass the whole command to a shell process run under sudo to have the file written to with root permissions, such as sudo sh -c "ls > /root/somefile".
* In a lot of office environments the ONLY local user on a system is Root. All other users are imported using NSS techniques such as nss-ldap. To setup a workstation, or fix it, in the case of a network failure where nss-ldap is broken, Root is required. This tends to leave the system unusable unless cracked. An extra local user, or an enabled Root password is needed here. The local user account should have its $HOME on a local disk, \_not\_ on NFS (or any other networked filesystem), and a .profile/.bashrc that doesn't reference any files on NFS mounts. This is usually the case for Root, but if adding a non-Root rescue account, you will have to take these precautions manually.
  + Alternatively, a sysadmin type account can be implemented as a local user on all systems, and granted proper sudo privileges. As explained in the benefits section above, commands can be easily tracked and audited.

# **Usage**

* When using sudo, your password is stored by default for 15 minutes. After that time, you will need to enter your password again.
* Your password will **not** be shown on the screen as you type it, not even as a row of stars (\*\*\*\*\*\*). It is being entered with each keystroke!

## sudo

To use sudo on the command line, preface the command with sudo, as below: *Example #1*

sudo chown bob:bob /home/bob/\*

*Example #2*

sudo /etc/init.d/networking restart

To repeat the last command entered, except with sudo prepended to it, run:

sudo !!

## Graphical sudo

You should **never** use normal sudo to start graphical applications as Root. You should use gksudo (kdesudo on *Kubuntu*) to run such programs. gksudo sets HOME=~root, and copies .Xauthority to a tmp directory. This prevents files in your home directory becoming owned by Root. (AFAICT, this is all that's special about the environment of the started process with gksudo vs. sudo).

Recent versions of some flavours might not have gksu installed.

If necessary install and set gksu-properties to sudo.

Examples:

gksudo gedit /etc/fstab

or

kdesudo kate /etc/X11/xorg.conf

* To run the graphical configuration utilities, simply launch the application via the Administration menu.
* gksudo and kdesudo simply link to the commands gksu and kdesu

## Drag & Drop sudo

This is a trick from [this thread](http://www.ubuntuforums.org/showthread.php?t=24008) on the [Ubuntu Forums](https://wiki.ubuntu.com/UbuntuForums).

Create a [launcher](https://help.ubuntu.com/community/HowToAddaLauncher) with the following command:

gksudo "gnome-open %u"

When you drag and drop any file on this launcher (it's useful to put it on the desktop or on a panel), it will be opened as Root with its own associated application. This is helpful especially when you're editing config files owned by Root, since they will be opened as read only by default with gedit, etc.

# **Users**

## Allowing other users to run sudo

To add a new user to sudo, open the **Users and Groups** tool from **System->Administration** menu. Then click on the user and then on properties. Choose the **User Privileges** tab. In the tab, find **Administer the system** and check that.

* In Hardy Heron and newer, you must first **Unlock**, then you can select a user from the list and hit **Properties**. Choose the **User Privileges** tab and check **Administer the system**.

Warning /!\ In the terminal (for Precise Pangolin, 12.04), this would be:

sudo adduser <username> sudo

where you replace <username> with the name of the user (without the <>).

In previous version of Ubuntu

sudo adduser <username> admin

would have been appropriate, but the admin group has been deprecated and no longer exists in Ubuntu 12.04.

## Logging in as another user

**Please don't use this to become Root,** see further down in the page for more information about that.

sudo -i -u <username>

For example to become the user amanda for tape management purposes.

sudo -i -u amanda

The password being asked for is your own, not amanda's.

## root account

### Enabling the root account

|  |  |  |
| --- | --- | --- |
| IconsPage/warning.png | **Enabling the Root account is rarely necessary. Almost everything you need to do as administrator of an Ubuntu system can be done via sudo or gksudo. If you really need a persistent Root login, the best alternative is to simulate a Root login shell using the following command...** | IconsPage/warning.png |

sudo -i

To enable the Root account (i.e. set a password) use:

sudo passwd root

**Use at your own risk!**

|  |  |  |
| --- | --- | --- |
| IconsPage/dont.png | **Logging in to X as root may cause very serious trouble.** If you believe you need a root account to perform a certain action, **please consult the official support channels first**, to make sure there is not a better alternative. | IconsPage/dont.png |

### Re-disabling your root account

|  |  |  |
| --- | --- | --- |
| IconsPage/info.png | If for some reason you have enabled your root account and wish to disable it again, use the following command in terminal... | IconsPage/info.png |

sudo passwd -dl root

# **Other Information**

## Misconceptions

* *Isn't sudo less secure than su?*
  + The basic security model is the same, and therefore these two systems share their primary weaknesses. Any user who uses su **or** sudo must be considered to be a privileged user. If that user's account is compromised by an attacker, the attacker can also gain root privileges the next time the user does so. The user account is the weak link in this chain, and so must be protected with the same care as Root.
  + On a more esoteric level, sudo provides some features which encourage different work habits, which can positively impact the security of the system. sudo is commonly used to execute only a single command, while su is generally used to open a shell and execute multiple commands. The sudo approach reduces the likelihood of a root shell being left open indefinitely, and encourages the user to minimize their use of root privileges.
* *I won't be able to enter single-user mode!*
  + The sulogin program in Ubuntu is patched to handle the default case of a locked root password.
* *I can get a root shell from the console without entering a password!*
  + You have to enter your password.
  + Console users have access to the boot loader, and can gain administrative privileges in various ways during the boot process. For example, by specifying an alternate init(8) program. Linux systems are not typically configured to be secure at the console, and additional steps (for example, setting a root password, a boot loader password and a BIOS password) are necessary in order to make them so. Note that console users usually have physical access to the machine and so can manipulate it in other ways as well.

## Special notes on sudo and shells

**None of the methods below are suggested or supported by the designers of Ubuntu.**

Please do not suggest this to others unless you personally are available 24/7 to support the user if they have issues as a result of running a shell as Root.

To start a *root shell* (i.e. a command window where you can run Root commands), starting Root's environment and login scripts, use:

sudo -i (similar to sudo su - , gives you roots environment configuration)

To start a *root shell*, but keep the current shell's environment, use:

sudo -s (similar to sudo su)

For a brief overview of some of the differences between su, su -, and sudo -{i,s} see : [Ubuntu Forums Post with nice table](http://ubuntuforums.org/showpost.php?p=6188826&postcount=4) .

Summary of the differences found -

corrupted by user's   
 HOME=/root uses root's PATH env vars  
sudo -i Y Y[2] N  
sudo -s N Y[2] Y  
sudo bash N Y[2] Y  
sudo su Y N[1] Y  
  
[1] PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games  
 probably set by /etc/environment  
[2] PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/X11R6/bin

For a detailed description of the differences see man su and man sudo .

# **Remove Password Prompt For sudo**

|  |  |  |
| --- | --- | --- |
| IconsPage/warning.png | **If you disable the sudo password for your account, you will seriously compromise the security of your computer. Anyone sitting at your unattended, logged in account will have complete Root access, and remote exploits become much easier for malicious crackers.** | IconsPage/warning.png |

* **This method is NOT suggested nor supported by the designers of Ubuntu.**
* Please do not suggest this to others unless you personally are available 24/7 to support the user if they have issues as a result of running a shell as Root.

These instructions are to remove the prompt for a password when using the **sudo** command. The **sudo** command will still need to be used for Root access though.

**Edit the sudoers file**

Open a Terminal window. Type in **sudo visudo**. Add the following line to the END of the file (if not at the end it can be nullified by later entries):

<username> ALL=NOPASSWD: ALL

Replace <username> with your user name (without the <>). This is assuming that Ubuntu has created a group with the same name as your user name, which is typical. You can alternately use the group *users* or any other such group you are in. Just make sure you are in that group. This can be checked by going to **System->Administration->Users and Groups**

Example:

michael ALL=NOPASSWD: ALL

Type in **^x** to exit. This should prompt for an option to save the file, type in **Y** to save.

Log out, and then log back in. This should now allow you to run the sudo command without being prompted for a password.

Or to do this for the system wide group **sudo**

root$ echo "%sudo ALL=(ALL) NOPASSWD: ALL" >> /etc/sudoers

Log out, and then back in.

# **Reset sudo timeout**

You can make sure sudo asks for password next time by running:

sudo -k

The default sudo timeout length can be changed by following this article: [RootSudoTimeout](https://help.ubuntu.com/community/RootSudoTimeout).

# **Other Resources**

* [fixing sudo](http://www.psychocats.net/ubuntu/fixsudo)
* [graphical sudo](http://www.psychocats.net/ubuntu/graphicalsudo)
* [Ubuntu Forums policy](http://ubuntuforums.org/showthread.php?t=1486138) on enabling the Root account
* [sudo](http://manpages.ubuntu.com/manpages/lucid/en/man8/sudo.8.html) man page
* [sudoers file](http://manpages.ubuntu.com/manpages/lucid/en/man5/sudoers.5.html) man page

[CategoryCommandLine](https://help.ubuntu.com/community/CategoryCommandLine) [CategorySecurity](https://help.ubuntu.com/community/CategorySecurity) [CategoryCommandLine](https://help.ubuntu.com/community/CategoryCommandLine)

RootSudo (last edited 2014-06-10 15:35:03 by [elfy](https://launchpad.net/%7Eelfy))

The material on this wiki is available under a free license, see [Copyright / License](https://help.ubuntu.com/community/License) for details

**You** can contribute to this wiki, see [Wiki Guide](https://help.ubuntu.com/community/WikiGuide) for details