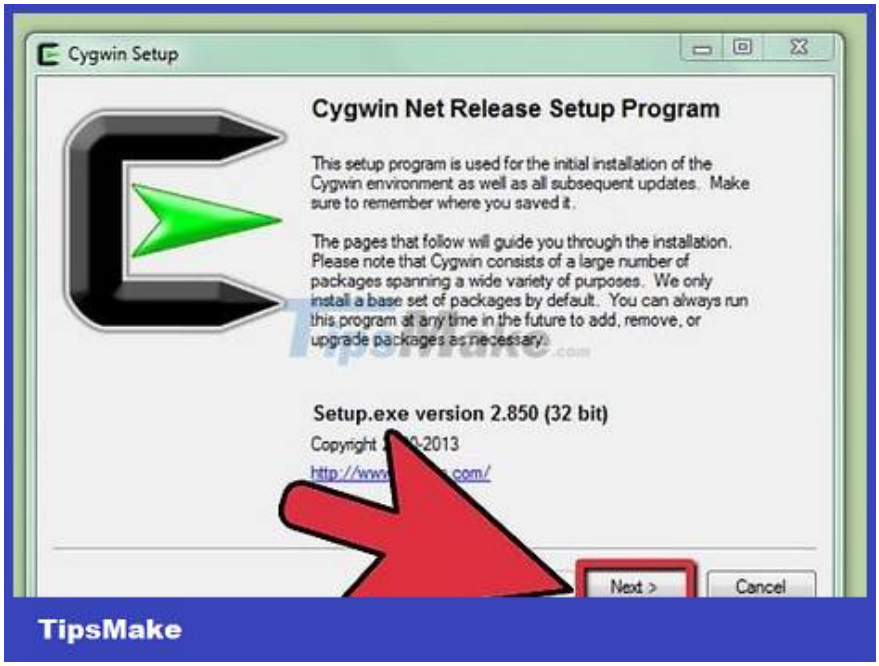
How to Use SSH

If you are connecting to another computer over the network, you probably want to keep your data safe. And SSH is a viable option. To do that, you need to properly set up SSH on your computer and then create an encrypted connection to the server. At the same time, SSH needs to be enabled at both ends of the connection. Follow the instructions below to ensure your connection is secure.

Connecting for the first time

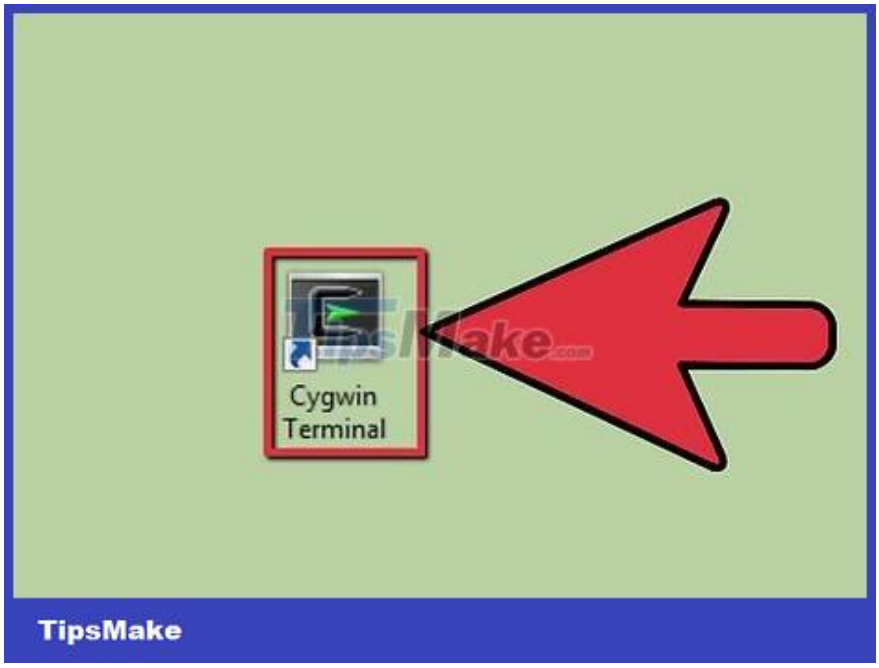


Install SSH. With Windows, you need to download and install the SSH client program. The most famous is Cygwin: you can download this program for free from the developer's website. Download and install as you would any other program. Besides Cygwin, PuTTY is also a popular free choice.

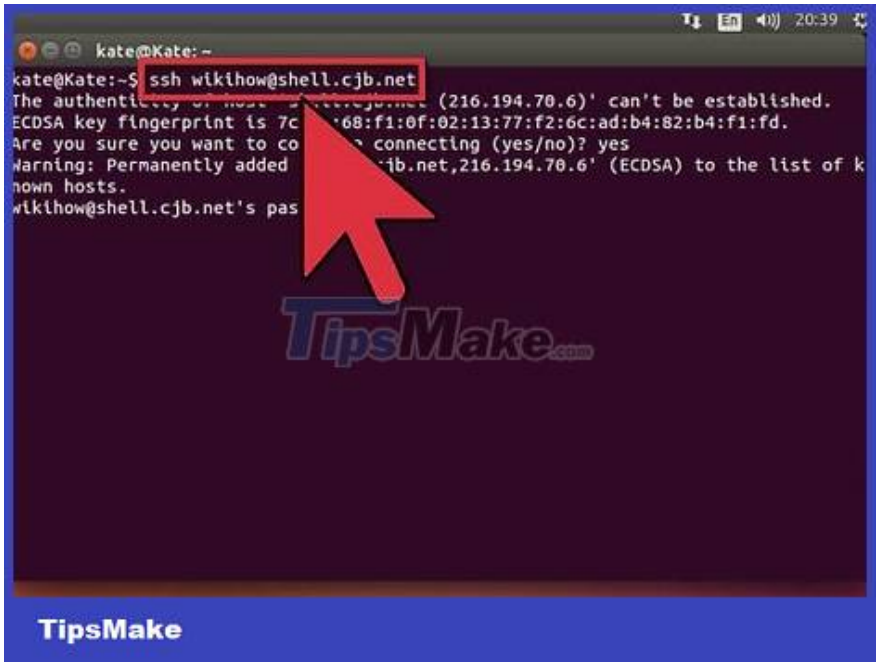
During the installation of Cygwin, you must choose to install OpenSSH from the Net section.

Linux and Mac OS X have SSH available. That's because SSH is a UNIX system and Linux and OS X were both developed from this system.

If you are using Windows 10 with the Anniversary Update, you can install Windows Subsystem for Linux: with this feature, SSH will be pre-installed.



Run SSH. Open the terminal program installed by Cygwin, or Bash on Ubuntu on Windows 10, or Terminal on OS X and Linux. SSH uses a terminal interface to interact with other computers. SSH doesn't have a graphical interface, so you'll have to get used to typing commands.



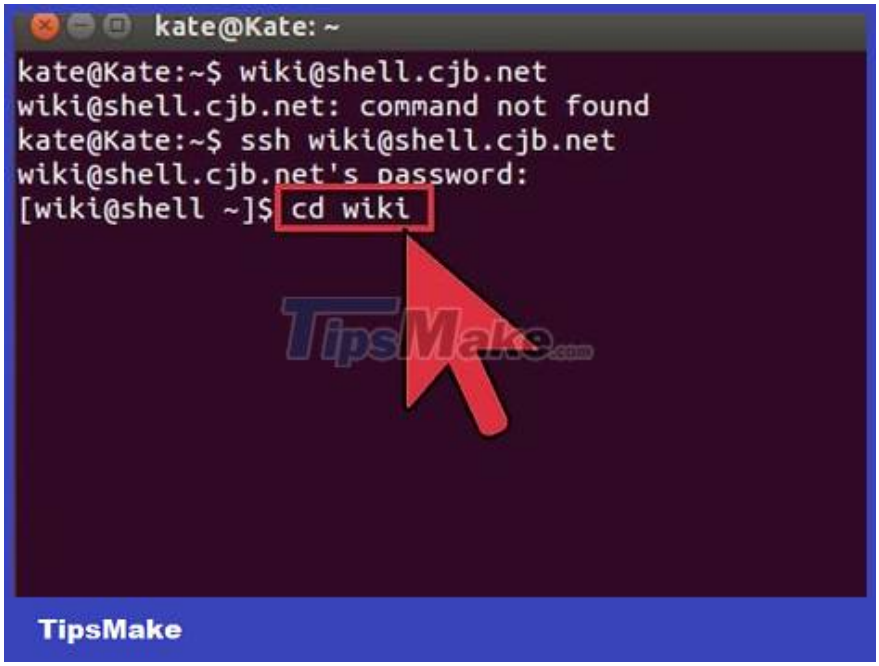
Checking connection. Before starting to create security keys and transfer files, you should check to make sure that SSH is configured correctly on the computer you are using as well as the other connection. Please enter the following command instead with your username on the other computer and by the address of that server or computer:

```
$ ssh  @
```

Once the connection is established, you will be asked to enter a password. As you type, the mouse pointer will not move and any characters entered will not be displayed.

If there is an error in this step, SSH has been misconfigured on your computer or the computer on the other end does not accept SSH connections.

Learn basic commands



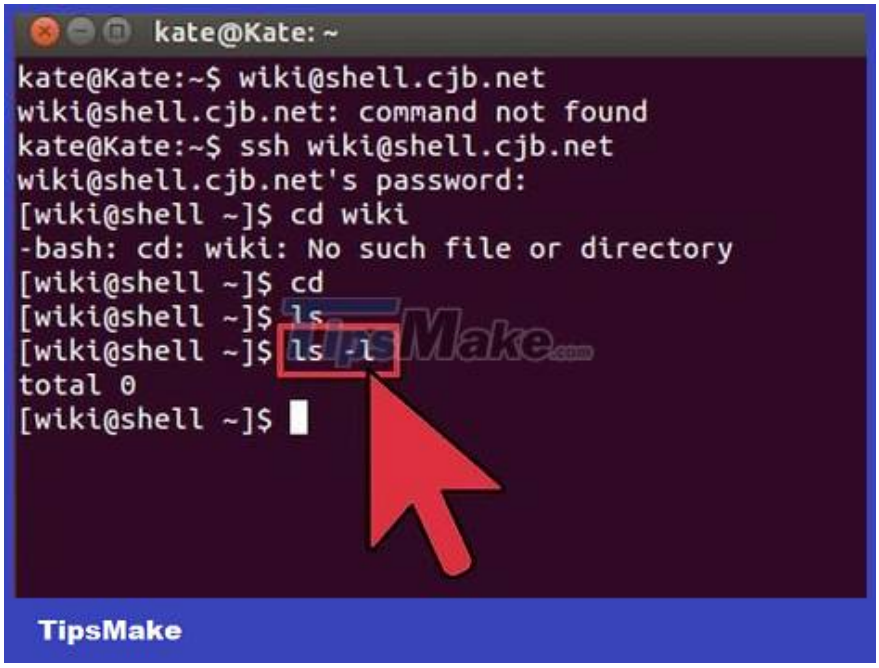
Navigate the SSH shell (command interpreter). During the first connection to the other computer, you should 'locate' to the HOME folder. To move within the directory structure, use the command `cd`:

`cd .` Move to the folder right in front of the directory tree

`cd` Move to any specific folder.

`cd /home/tên_th?_m?c/???ng_d?n/` Move to a specific folder from the root folder (home).

`cd ~` return to your HOME directory.

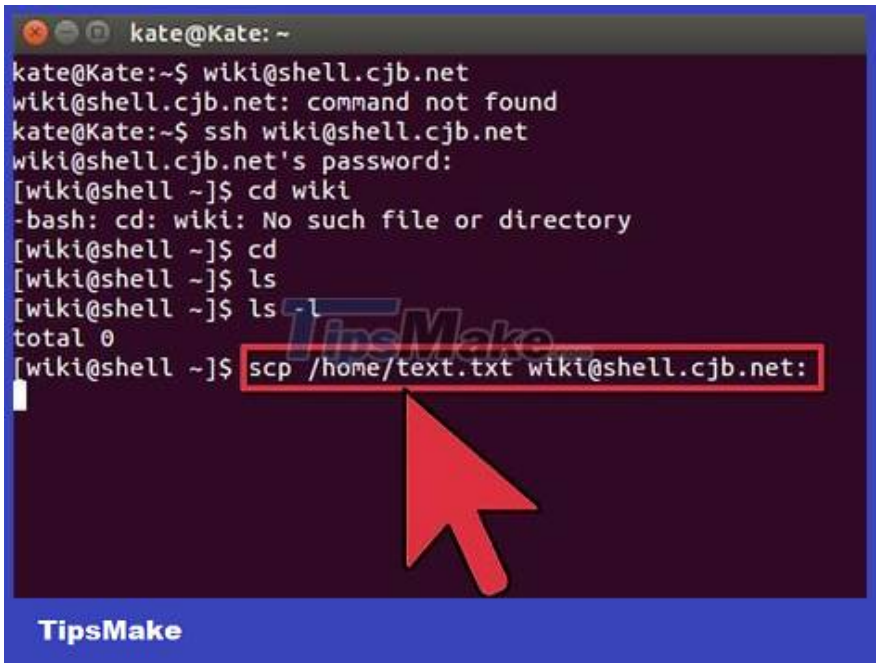


Check the content contained in the current directory. To view files and folders in the current directory, you can use the command `ls`:

`ls`lists every directory and file in the current directory.

`ls -l`lists the contents of the folder along with additional information such as size, permissions, and date.

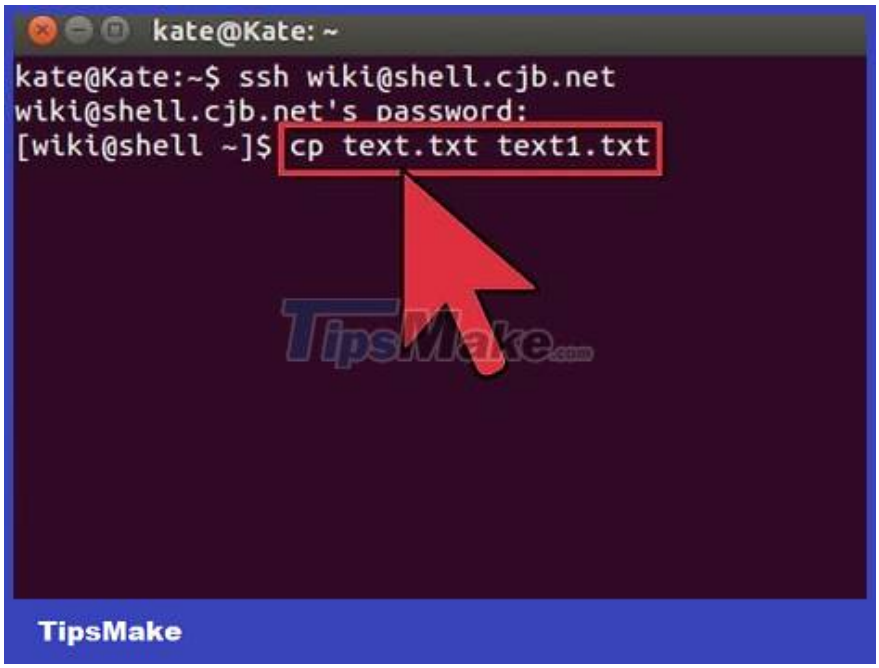
`ls -a`Lists all contents, including hidden files and folders.



Copy files from your location to the other computer. To copy files from the computer you are using to the computer you are accessing remotely, you can use the command `scp`:

`scp /th?_m?c_c?c_b?/ví_d?_1.txt @ :???ng_d?n>` will copy example1.txt to a specific on the computer being accessed remotely. You can leave the field blank to copy to the root directory of this computer.

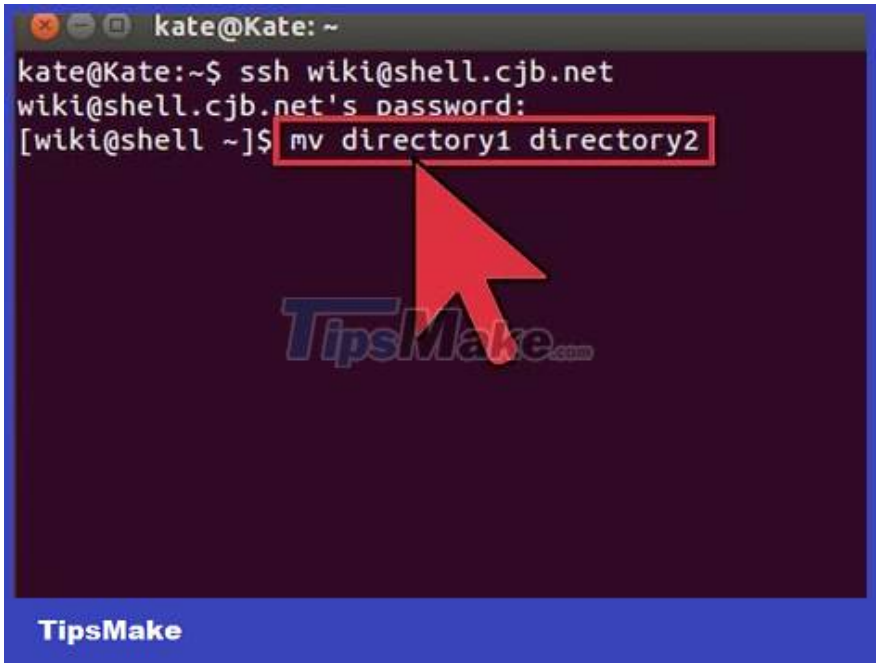
`scp @ :/home/ví_d?_1.txt ./` will move example1.txt from the home directory on the remote computer to the directory you are entering directly on this head machine.



Copy files through shell. You can use the command `cp`to copy files in the same directory or to a predetermined directory:

`cp ví_d?_1.txt ví_d?_2.txt`will create a copy of example1.txt and name it example2.txt right in the current directory.

`cp ví_d?_1.txt th?_m?c/`will create a copy of example1 in the directory specified by directory.



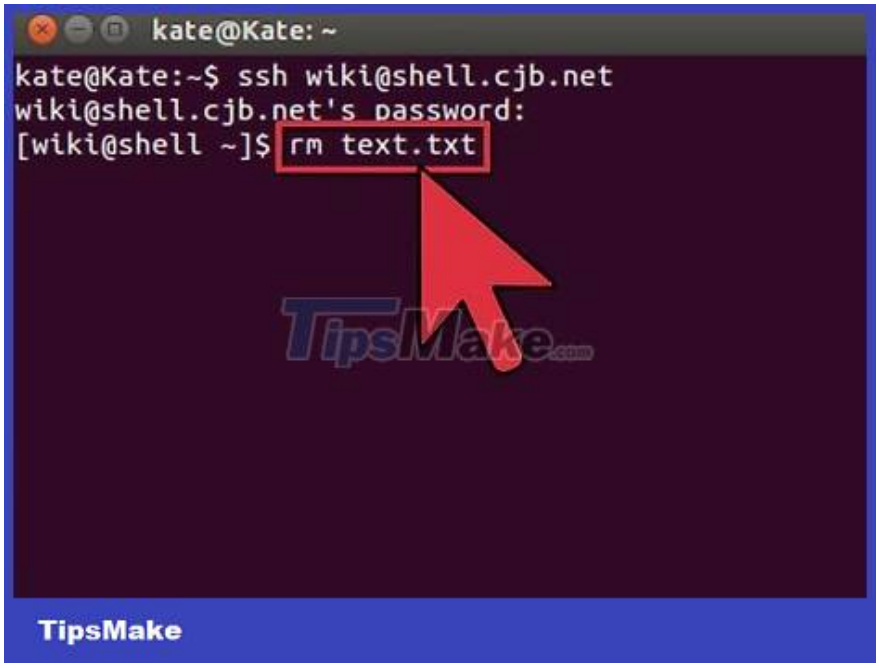
Transfer and rename files. If you want to rename or move (without copying) certain files, you can use the command `mv`:

`mv ví_d?_1.txt ví_d?_2.txt`will rename example_1.txt to example_2.txt, the file is still in the old directory.

`mv th?_m?c_1 th?_m?c_2`rename folder_1 to folder_2. The content contained in the folder remains unchanged.

`mv ví_d?_1.txt th?_m?c_1/`move example1.txt to folder1.

`mv ví_d?_1.txt th?_m?c_1/ví_d?_2.txt`Move example1.txt to folder1 and rename it to example2.txt.

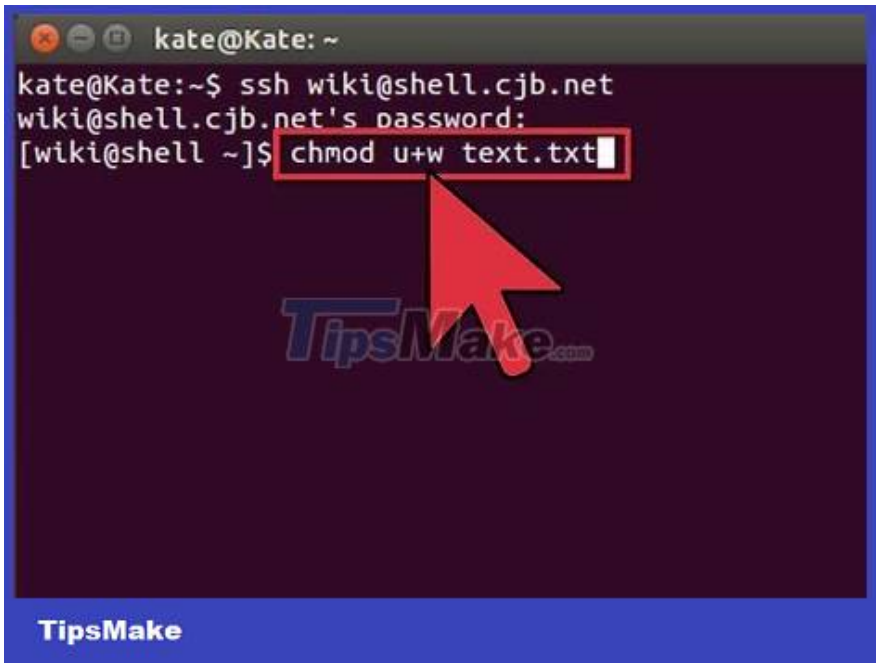


Delete files and folders. To delete content on a computer being accessed remotely, you can use the command `rm`:

`rm ví_d?_1.txt` delete the file named example_1.txt.

`rm –I ví_d?_1.txt` Delete file example1.txt after receiving confirmation from you.

`rm th?_m?c_1/`delete folder_1 with all its contents.

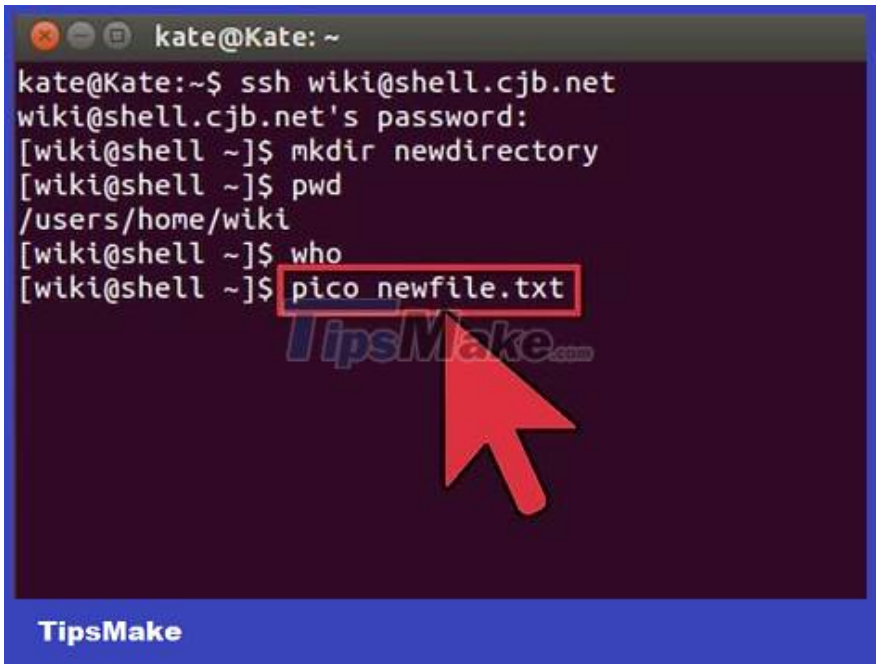


Change permissions of files. You can change the read and write permissions of a file with the command `chmod`:

`chmod u+w ví_d?_1.txt`add permission to write (modify) files to user (u). You can also use plugins `g`for group permissions and `o`for world permissions.

`chmod g+r ví_d?_1.txt`Add permission to read (access) files to the group.

The list of commands you can use to secure or unlock various aspects of your device is quite long.



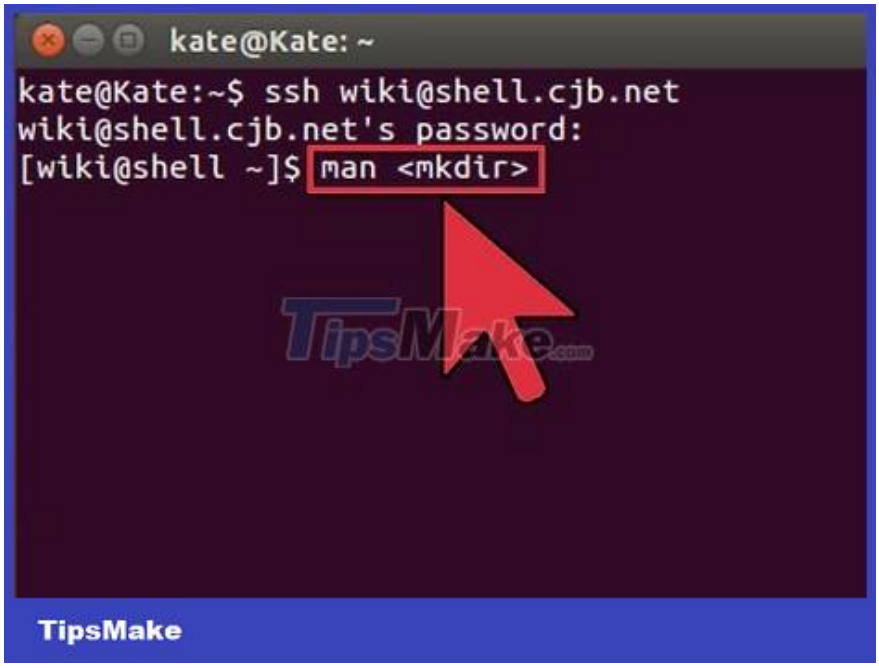
Learn other miscellaneous basic commands. There are still a few more important commands that are used quite a lot on the shell interface, including:

`mkdir th?_m?c_m?i`create a subfolder called new_folder.

`pwd`displays the current directory location.

`who`Displays who is logging into the system.

`pico newfile.txt`or `vi newfile.txt`create a new file and open a file editor. Different machines may have different file editors installed. Pico and vi are the two most popular programs. In case your computer uses another file editor, you may have to use other commands.

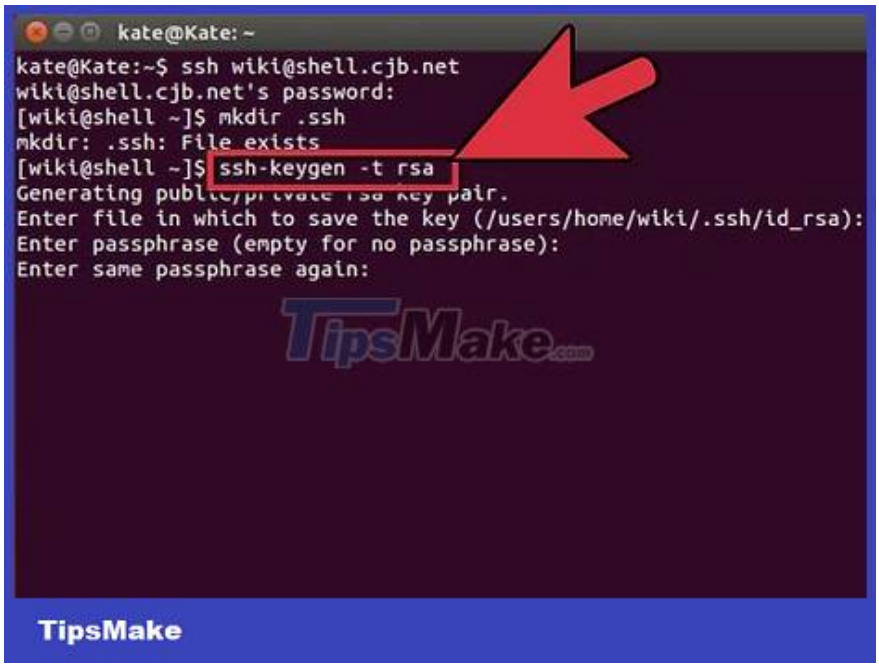


Get detailed information about any command. If you're not sure what a command does, you can use it `man`to learn about all the parameters and possible uses:

`man` displays information about that command

`man -k` finds all command pages for the specified keyword.

Generate encrypted keys



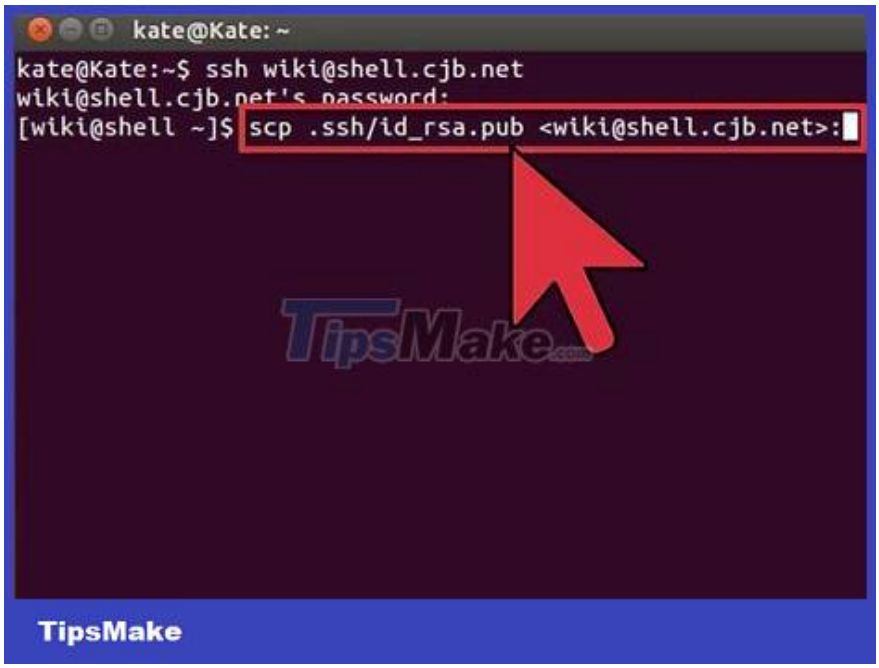
Generate your SSH key. These keys allow you to connect to a remote device without having to enter a password each time you connect. This method is much safer because with it, you do not need to send a password over the network.

Create a locked folder on your computer by entering the command`$ mkdir .ssh`

Generate private and public keys using the command`$ ssh-keygen –t rsa`

You will be asked if you want to create a password for the key: it is not required. If you don't want to create a password, just press Enter. Two keys id_rsa and id_rsa.pub will be created in the .ssh directory.

Change private key permissions. To ensure that only you can read the private key, enter the command`$ chmod 600 .ssh/id_rsa`

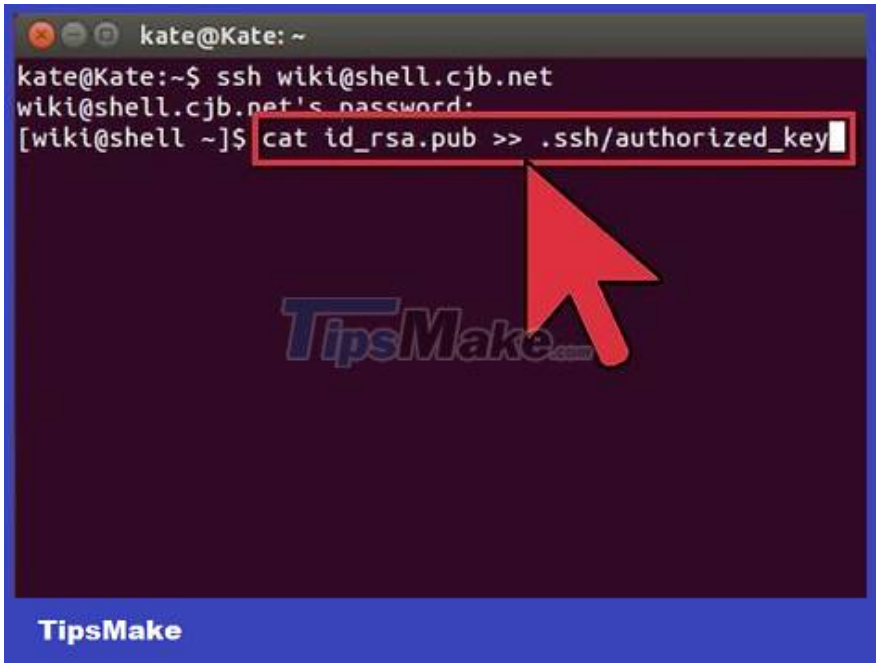


Leave the public key on the other computer. Once you've created the key, you're ready to place the public key on the other end of the connection so you can connect without a password. Enter the following command, replacing the necessary parts as shown:

`$ scp .ssh/id_rsa.pub  @  :`

Don't forget the colon (:) at the end of the command.

You will be asked to enter your password before starting the file transfer.

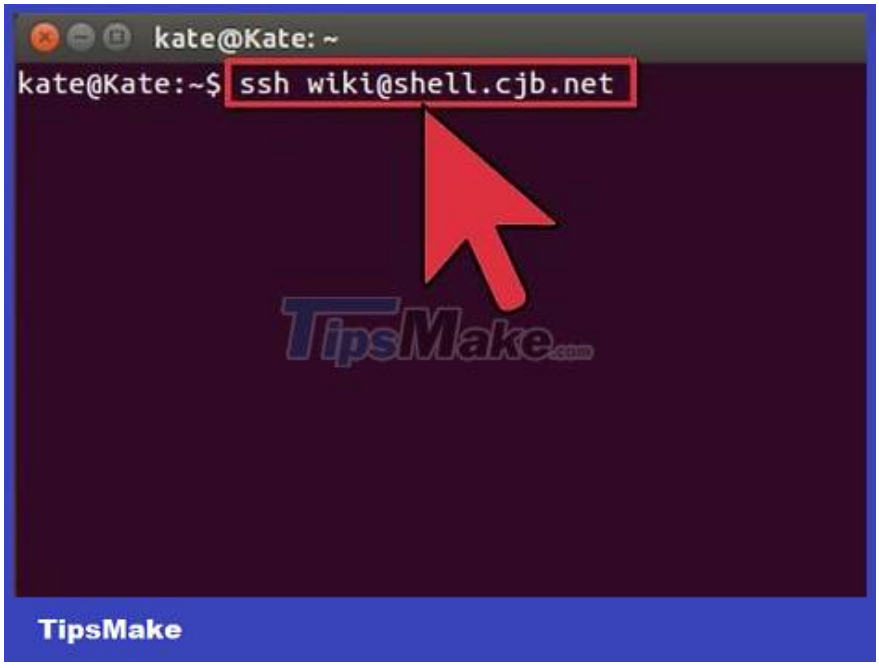


Install the public key on the other computer. Once set, you need to install the lock on the other machine for it to work properly. First, log in to the other computer the same way you did in step 3.

Create an SSH directory if it doesn't already exist on this computer:`$ mkdir .ssh`

Mount your key with the licensed key file. If this file does not already exist, it will be initialized:`$ cat id_rsa.pub >> .ssh/authorized_keys`

Change the permissions for the SSH directory to allow access:`$ chmod 700 .ssh`



Check if the connection is working or not. After installing the lock on the computer at the other end of the connection, you should be able to create a connection without having to enter a password. Use the following

command to check your connection:`$ ssh  @`

If you don't have to enter a password when connecting, the lock must have been configured correctly.

You finished reading the article "**How to Use SSH**" edited by the **TipsMake** team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.
