

How to store your own Minecraft server

Minecraft is still a very popular game and an attractive part behind this game is the ability to store and run your own servers. Storing a Minecraft server is very simple and you can run it quickly.

Minecraft is still a very popular game and an interesting part of this game is the ability to store and run your own servers. Storing a Minecraft server is very simple and you can run it quickly.

This tutorial will show you one of the many ways to run Minecraft server. This method is one of the simplest and most stable server settings possible at the present time.

Instructions for storing personal Minecraft servers

1. Before start
2. Install dependencies
3. Download Minecraft server
4. Write a startup script
5. Start the server

Before start

If you specify to run Minecraft server on the local network, this is not a problem, but if you want people to be able to play on your server over the Internet, you will need to find the appropriate hosting service for the server.

There are many great options that you can use to store Minecraft server. **Linode** and **DigitalOcean** are often a safe option. You will need a VPS (Virtual Private Server) to host Minecraft. You can store on cheap shared hosting, often designed to host simple websites.

In addition, you should also host the server on Linux. Although Minecraft server can be run on Windows, generally, storing on Linux will be cheaper and maintenance is easier. Ubuntu is one of the reliable Linux distributions. It's quite friendly to beginners, stable and has an active community to help if you need it.

Everything here assumes that you are hosting and have access to a terminal, via SSH or the web interface provided by your server. Any good VPS server will allow you to access the terminal.

Install dependencies

```
The following additional packages will be installed:
ca-certificates-java default-jdk-headless default-jre default-jre-headless
fonts-dejavu-extra java-common libatk-wrapper-java libatk-wrapper-java-jni
libice-dev libpthread-stubs0-dev libsm-dev libutempter0 libx11-dev
libx11-doc libxau-dev libxcb1-dev libxdmcp-dev libxt-dev openjdk-11-jdk
openjdk-11-jdk-headless openjdk-11-jre openjdk-11-jre-headless
x11proto-core-dev x11proto-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
default-java-plugin libice-doc libsm-doc libxcb-doc libxt-doc
openjdk-11-demo openjdk-11-source visualvm fonts-ipafont-gothic
fonts-ipafont-mincho fonts-wqy-microhei | fonts-wqy-zenhei byobu | screenie
| iselect
The following NEW packages will be installed:
ca-certificates-java default-jdk default-jdk-headless default-jre
default-jre-headless fonts-dejavu-extra java-common libatk-wrapper-java
libatk-wrapper-java-jni libice-dev libpthread-stubs0-dev libsm-dev
libutempter0 libx11-dev libx11-doc libxau-dev libxcb1-dev libxdmcp-dev
libxt-dev openjdk-11-jdk openjdk-11-jdk-headless openjdk-11-jre
openjdk-11-jre-headless screen x11proto-core-dev x11proto-dev
xorg-sgml-doctools xtrans-dev
0 upgraded, 28 newly installed, 0 to remove and 0 not upgraded.
Need to get 128 MB of archives.
After this operation, 302 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```



You will need to install some software packages before you can run Minecraft server. You can install them directly with the Ubuntu Apt package manager. Start by running the following command in the terminal on your server:

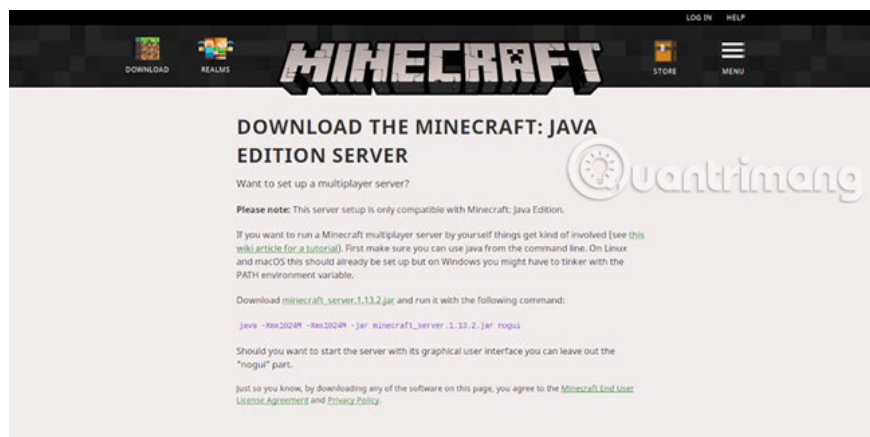
```
sudo apt install default-jdk screen wget
```

If you have never used a Linux package manager before, wait a few seconds while Ubuntu installs the new software. It will notify you when the installation is finished.

Download Minecraft server

Set up the directory where you want to run the server. This is not very important. You can do everything in your Home folder, if you feel it is the most convenient for you.

```
mkdir ~/Minecraft cd ~/Minecraft
```



On the computer you are using, download the Minecraft server. Find the download link for the latest version of Minecraft server and copy it.

Go back to the terminal server and start typing the line below:

```
wget -O minecraft_server.jar
```



```
nick@ubuntuBionic:~/Minecraft$ wget -O minecraft_server.jar https://launcher.mojang.com/v1/objects/3737db93722a9e39eeada7c27e7aca28b144ffa7/server.jar
--2018-12-09 23:18:33-- https://launcher.mojang.com/v1/objects/3737db93722a9e39eeada7c27e7aca28b144ffa7/server.jar
Resolving launcher.mojang.com (launcher.mojang.com)... 216.137.63.16
Connecting to launcher.mojang.com (launcher.mojang.com)|216.137.63.16|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 33835116 (32M) [application/java-archive]
Saving to: 'minecraft_server.jar'

minecraft_server.jar 3%[          ] 1.16M  459KB/s
```

Paste the address you copied above here. Everything will look like this:

```
wget -O minecraft_server.jar https://launcher.mojang.com/v1/objects/3737db93722a9e39eeada7c27e7aca28b144ffa7/server.jar
```

You will need to accept the Minecraft license to run the server. You do that by creating a file in the same directory as the server. The server will read the file and see that you have accepted the license.

```
echo "eula=true" > eula.txt
```

Write a startup script

In fact, you don't need a startup script, but it's easier to combine everything into a script, then you just need to run a command to start your server. Start creating a new file by opening it in the text editor. If you are not familiar with Linux text editors, use **Nano**.

```
nano start-server.sh
```

Then, enter the following code in the script:

```
#!/bin/bash /usr/bin/screen -S $1 /usr/bin/java -Xmx1024M -Xms1024M -jar minecraft_server.jar
```

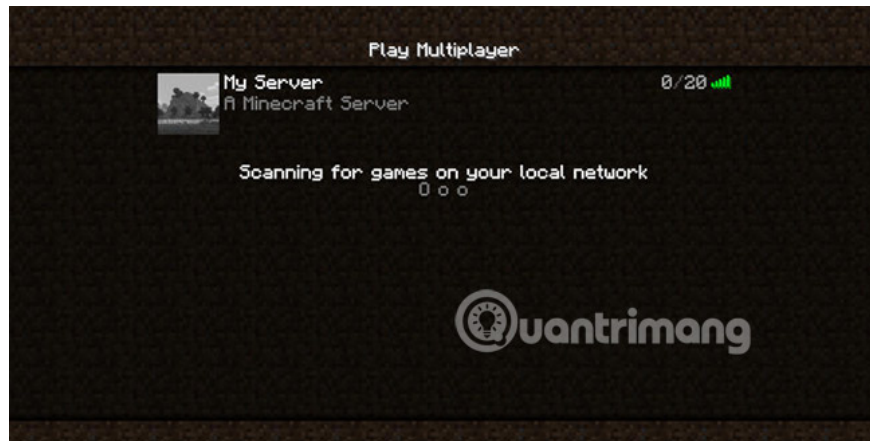
Save the script and exit the text editor. You also need to make your script run before running it.

```
chmod +x start-server.sh
```

Start the server

Finally, your server is ready to boot. Give the server a name so you can easily identify it when you run the script. (The main server name is the **'New-Server' section**).

```
./start-server.sh "New-Server"
```



The server will start and you can connect by entering your server IP address into the Minecraft client. Remember to keep the server up to date by replacing the server's **.jar** file with new releases.

If you plan to make your server public, then you should learn tips about hosting VPS to better secure it. Activating a firewall is also a consideration. Ubuntu has a great option in the form of UFW.

Good luck!

See more:

1. Instructions for setting up and managing FTP Server on Windows 10
2. Instructions for setting up individual FTP Server with FileZilla
3. How to build a game server on Linux

You finished reading the article "**How to store your own Minecraft server**" 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.