



# Cross Compiling for windows from Ubuntu 18.04

*Based on **Northern Coin** project*


# Premise

I'm not a Dev, I don't know much about Linux and everything you're going to read in this guide is what I've learned through the web and put it together, maybe there is better ways to do it, maybe I'm also writing some not really accurate or even wrong things, they worked for me in many occasions. So take this guide as a main path to follow that may need to be modified depending on the case.

I'm going to use the **Northern Coin** project as example but this method should work for different coin wallets too.

## Introduction

In the following guide you will find step by step the code to put in the command prompt on Ubuntu 18.04 but should work also on other versions of Linux based operating system.

To work properly you have to copy one line at the time, paste it in the command line and run it by pressing the "return" key. The beginning of the line is marked with  and due to length reasons, here on the guide, sometime one line goes split in two lines, you have to copy both at once.

One last note, normally in every wallet git repository you find a folder that is usually called "doc" where you can find instructions on how to compile a wallet for different operating system.

### Note

If you are using Ubuntu on Windows, remember to run it as Administrator.



# The Basics

A good way to start no matter in which system are you working is to be up to date with programs and dependencies.

To update on Linux systems usually input the following code lines:

```
➤ sudo apt update  
➤ sudo apt upgrade
```

Sometimes you will be asked to chose if you want to update some packages and answer with Y or N

## Note

Compiling wallets very often require specifics versions of packages, and if you update them they will give you errors. Some of the most common are libtool and libdb4.8-dev witch if they are a newer version than required they will not work.

At this point we will install the dependencies needed to compile our wallet

## For the Northern wallet in general:

```
➤ sudo apt install build-essential libtool autotools-dev autoconf pkg-config libssl-dev
```

Then we need to add the db4.8 packages

```
➤ sudo add-apt-repository ppa:bitcoin/bitcoin  
➤ sudo apt-get update  
➤ sudo apt-get install libdb4.8-dev libdb4.8++-dev
```

Next we install the packagies for creating the qt (GUI wallet)

```
➤ sudo apt-get install libqt5gui5 libqt5core5a libqt5dbus5 qttools5-dev qttools5-dev-tools  
➤ sudo apt-get install libprotobuf-dev protobuf-compiler
```

## For the Cross-Compilation:

```
➤ sudo apt install automake bsdmainutils curl git
```

If you want to build the windows installer with make deploy you need NSIS:

```
➤ sudo apt install nsis
```

# Building for Windows 64-bit

At this point we are ready to start the cross compilation process.  
Download the wallet source code:

```
➤ cd
➤ git clone https://github.com/northern-community/Northern.git
```

Enter the newly created folder:

```
➤ cd Northern
```

The first step is to install the mingw-w64 cross-compilation tool chain:

```
➤ sudo apt install g++-mingw-w64-x86-64
➤ sudo update-alternatives --config x86_64-w64-mingw32-g++
```

Set the default mingw32 g++ compiler option to Auto hitting 0

Then we continue with:

```
➤ PATH=$(echo "$PATH" | sed -e 's:/mnt.*//g')
```

## Note

This line strip out problematic Windows %PATH% imported var that you will probably have if you are compiling with Ubuntu on Windows using Subsystem for Linux. It is probably good to do it also if you are compiling on a native Linux environment, if it gives you problems reboot the system and it should work.

Then copy and paste line by line the following code:

```
➤ cd depends
➤ make HOST=x86_64-w64-mingw32
➤ cd ..
➤ ./autogen.sh
➤ CONFIG_SITE=$PWD/depends/x86_64-w64-mingw32/share/config.site ./configure --with-gui=qt5
  --prefix=/
```

## Note

It is possible that when you copy a line witch is split on two lines the soft will add a double space where it is split. When you paste, check if between "--with-gui=qt5" and "--prefix=/" there is only one space.

And last the make command:

```
➤ make
```

#### Note

If you have a system with a good amount of ram you can set how many core to use while compiling the wallet with -jN where N is the number of core. Remember that the more core you use the more ram is needed, but the compile process will take far less time.

As example: make -j4

At the end of the process you will find the compiled file in the following folders.

northern-cli, northernd and northern-tx in:

**Northern/src**

while the northern-qt in:

**Northern/src/qt**

All those file will be needed and you have to put them all in the same folder in your Windows system. To start the wallet double click on northern-qt.exe.

## **Northern Coin** project

<http://nortcrypto.com>

<https://discord.gg/9nzt37V>

<https://github.com/northern-community>

Hope you found this guide useful and if you like it  
then feel free to DONATE to:

NORT: NPmk7ioBy7XzhZhZ1X4MnxVYXz6yDcA14y

BTC: 17f1xFseXNwLp9xBAP7iAygLFY8aiM22mP



Galapagos#7937