

Installing Clang on Ubuntu: A Step-by-Step Guide

TechRounder PDF Edition

Live article: <https://www.techrounder.com/how-to/installing-clang-on-ubuntu-a-step-by-step-guide/>

By Vipin PG | Published August 30, 2024 | Updated December 13, 2025 | Format: Guide | 5 min read

Quick answer

To install Clang on Ubuntu, open a terminal and run `sudo apt update` followed by `sudo apt install clang`, then verify the installation with `clang --version`. For a specific version, download the LLVM install script using `wget https://apt.llvm.org/llvm.sh`, make it executable with `chmod +x llvm.sh`, and run `sudo ./llvm.sh`.

Clang is a powerful tool for programmers working with C, C++, and Objective-C languages. It's an alternative to the more commonly used GNU Compiler Collection (GCC). If you're using Ubuntu and want to start working with Clang, you've come to the right place. This article will walk you through the process of installing Clang on your Ubuntu system, whether you're using version 20.04, 22.04, or another Ubuntu release.

What is Clang?

Before we dive into the installation process, let's take a moment to understand what Clang is and why you might want to use it. Clang is a compiler front-end, which means it's the part of the compiler that reads and understands your code. It's known for:

1. Fast compilation times
2. Helpful error messages
3. Good support for the latest language standards
4. Working well on different computer systems

Many programmers choose Clang because it can make their work easier and more efficient.

Preparing Your Ubuntu System

Before we install Clang, we need to make sure your Ubuntu system is ready. Here's what you need to do:

Updating Your System

It's always a good idea to update your system before installing new software. This ensures you have the latest security updates and that everything will work smoothly. To update your system, open a terminal window. You can do this by pressing `Ctrl + Alt + T` on your keyboard. Then, type the following command and press `Enter`:

```
sudo apt update
```

You might be asked for your password. Type it in (you won't see the characters as you type) and press `Enter`. After the update process finishes, run this command to install any available upgrades:

```
sudo apt upgrade
```

Press `Y` and `Enter` if asked to confirm the upgrades.

Installing Clang

Now that your system is up to date, we can install Clang. Ubuntu makes this process easy.

Basic Installation

For most users, the basic installation will be enough. Here's how to do it:

1. Open your terminal if it's not already open.
2. Type the following command and press Enter: 'sudo apt install clang'
3. If asked to confirm, type Y and press Enter.

That's it! Clang is now installed on your system.

Checking the Installation

To make sure Clang installed correctly, we can check its version. In your terminal, type:

```
clang --version
```

This will show you which version of Clang you have installed.

Using Clang

Now that you have Clang installed, you might be wondering how to use it. Here are some basic examples:

Compiling a C Program

Let's say you have a simple C program called "hello.c" that looks like this:

```
#include <stdio.h>

int main() {
    printf("Hello, World!\n");
    return 0;
}
```

To compile this program using Clang, you would use this command:

```
clang hello.c -o hello
```

This tells Clang to compile "hello.c" and create an output file called "hello".

Compiling a C++ Program

For a C++ program named "hello.cpp", you would use:

```
clang++ hello.cpp -o hello
```

The 'clang++' command is used for C++ programs.

Advanced Clang Features

Clang offers more than just basic compilation. Here are some advanced features you might find useful:

Static Analysis

Clang can analyze your code for potential problems without actually running it. This is called static analysis. To use this feature, add the '--analyze' flag when compiling:

```
clang --analyze hello.c
```

Optimization Levels

Clang can optimize your code to make it run faster. There are different levels of optimization, from -O0 (no optimization) to -O3 (aggressive optimization). For example:

```
clang -O2 hello.c -o hello
```

This uses level 2 optimization, which is a good balance between compile time and program performance.

Installing Specific Versions of Clang

Sometimes, you might need a specific version of Clang for your project. Here's how you can install different versions:

1. First, add the LLVM repository to your system. LLVM is the project that Clang is part of. Run these commands: 'wget https://apt.llvm.org/llvm.sh chmod +x llvm.sh'
 2. Now you can install a specific version. For example, to install Clang 16: 'sudo ./llvm.sh 16'
- Replace "16" with the version number you need.

Uninstalling Clang

If you ever need to remove Clang from your system, you can do so with this command:

```
sudo apt remove clang
```

This will remove the basic Clang package. If you installed additional components, you might need to remove those separately.

Troubleshooting Common Issues

Even with a straightforward installation process, you might run into some issues. Here are some common problems and how to solve them:

"Command not found" Error

If you get a "command not found" error when trying to use Clang, it might not be in your system's PATH. Try running:

```
export PATH="/usr/local/bin:$PATH"
```

Then try using Clang again.

Compilation Errors

If you're getting unexpected errors when compiling your code, make sure you're using the right command for your programming language (clang for C, clang++ for C++).

Version Conflicts

If you need to use multiple versions of Clang, you can install them side by side and use version-specific commands like 'clang-16' for Clang version 16.

Clang vs GCC: What's the Difference?

You might be wondering how Clang compares to GCC, another popular compiler. Here are some key differences:

1. Error Messages: Clang is known for more user-friendly error messages, which can help you find and fix problems in your code more quickly.
2. Compilation Speed: Clang often compiles code faster than GCC, especially for C++ projects.
3. Standard Compliance: Both Clang and GCC strive to comply with language standards, but they might interpret some edge cases differently.
4. Platform Support: GCC supports more platforms and architectures, but Clang is catching up quickly.

Best Practices When Using Clang

To get the most out of Clang, consider these best practices:

1. Keep Clang Updated: Regularly update Clang to benefit from bug fixes and new features.
2. Use Warning Flags: Compile with warning flags like '-Wall' and '-Wextra' to catch potential issues early.
3. Explore Clang Tools: Clang comes with various tools for code analysis and formatting. Explore tools like clang-tidy and clang-format to improve your code quality.
4. Read the Documentation: Clang has extensive documentation. When in doubt, refer to the official Clang documentation for detailed information.

Conclusion

Installing and using Clang on Ubuntu is a straightforward process that can greatly improve your programming experience. Whether you're a seasoned developer or just starting out, Clang's fast compilation times and helpful error messages can make your coding journey smoother. Remember, the key steps are:

1. Update your system
2. Install Clang using 'sudo apt install clang'
3. Verify the installation with 'clang --version'

From there, you can start compiling your C, C++, or Objective-C programs with Clang. As you become more comfortable with it, you can explore its advanced features to optimize your code and catch potential issues early. Happy coding with Clang on Ubuntu!