NEWS · AUGUST 25, 2025

# How to configure Linux for Delphi development

By Dr. Holger Flick

Delphi has long been a powerful tool for Windows application development, but with the introduction of Linux support in recent versions, developers can now leverage Delphi's capabilities to create applications for Linux environments. This opens up new opportunities for cross-platform development and expands the reach of Delphi applications.

The difficulty for many developers lies in the configuration of Linux that you can connect it to RAD Studio so that you can deploy your Linux applications directly from the IDE. This process involves several steps, configuring the Linux target in RAD Studio, and ensuring that all necessary libraries and dependencies are in place.

As always, you find great resources on GitHub from other Delphi developers.

## Why do you mention Linux?

Linux is a popular choice for many developers due to its flexibility, stability, and open-source nature. By supporting Linux, Delphi developers can reach a broader audience and take advantage of the strengths of both Windows and Linux platforms. Additionally, Linux is widely used in server environments, making it an attractive option for developing backend applications and services.

Just to give you an example, I run all my web servers on Linux and I also run all my XData backend services on Linux. I simply love the stability and performance of Linux servers. Installing reverse proxies like Nginx or Apache is a breeze and does not require a lot of resources. And best of all: Linux is free! Renting a Linux server is often cheaper than 5 bucks a month. You won't find a Windows server for that price.

## A note on Delphi developers and GitHub

It's common to hear that Delphi developers are less active on platforms like GitHub compared to other programming communities. While open-source collaboration is widespread elsewhere, Delphi developers often hesitate to share their code publicly or contribute to shared projects. This reluctance can limit opportunities for learning, collaboration, and innovation. Embracing platforms like GitHub can help Delphi developers connect with peers, improve code quality, and contribute to the growth of the community.

Still, talking to many Delphi developers, I often hear that they are not familiar with GitHub or do not see the value in sharing their code on a platform that they cannot control. This is a missed opportunity for the Delphi community to grow and thrive.

## Version control? I only use backups and ZIP!

Sadly, another frequent occurence when talking to Delphi developers is that they do not use version control systems like Git. Many developers rely on manual backups or ZIP files to manage their code, which can lead to issues with code management, collaboration, and tracking changes over time. Version control systems like Git provide a robust way to manage code changes, collaborate with other developers, and maintain a history of code changes. By using Git, Delphi developers can improve their development workflow, reduce the risk of code loss, and enhance collaboration within their teams. It is also **very** easy to embrace as the learning curve is not steep at all. And best of all: Git is free! And one of the best books is free as well: Pro Git.

> Please use version control!

## Back to Linux

To get started with Delphi on Linux, you'll need to set up a Linux environment that RAD Studio can connect to. This typically involves installing a compatible Linux distribution, configuring SSH access, and ensuring that the necessary libraries and tools are installed on the Linux machine.

But nowadays, Windows even offers a Windows Subsystem for Linux (WSL) that you can use to run a Linux environment directly on your Windows machine. This can simplify the setup process and make it easier to test and deploy your Delphi applications on Linux.

Once your Linux environment is set up, you can configure RAD Studio to connect to it. This involves adding a new Linux target in the IDE, specifying the connection details, and testing the connection to ensure everything is working correctly.

## How to configure Linux for RAD Studio

Ian Barker and Jim McKeeth to the rescue! Initally, they created a great series of Gists that walk you through the entire process of setting up Linux for RAD Studio. As Gists are a bit hard to find, I have copied the content to a GitHub repository that you can find here:

- Delphi on Linux Setup

This repository contains detailed instructions and scripts to help you configure your Linux environment for Delphi development. It covers everything from installing the necessary packages to configuring SSH access and setting up the RAD Studio connection.

## Surf GitHub for more resources

There's so many great resources on GitHub from other Delphi developers. Just search for "Delphi Linux" or similar terms to find repositories, gists, and projects that can help you get started with Delphi on Linux. You can find sample projects, libraries, and tools that can make your development process easier and more efficient.

## Summary

Running Delphi applications on Linux is now more accessible than ever, thanks to the support provided in recent versions of RAD Studio. By setting up a Linux environment and configuring RAD Studio to connect to it, developers can take advantage of Delphi's powerful features to create cross-platform applications. This opens up new opportunities for reaching a broader audience and leveraging the strengths of both Windows and Linux platforms.

## Ready to expand what your Delphi applications can do?

Whether it’s deploying to Linux, building web APIs, or learning new frontend tools, I’ll guide you through planning, setup, and delivery. Reach out and we’ll map the next steps together.

INDEPENDENT · AD-FREE · READER-SUPPORTED

# Free to read. Not free to make.

Every tutorial, article, and line of example code on Holger's Code is published without a paywall and without ads. This page is about what keeps it that way.

## Professional work, published free

The content here is held to the same standard as professional client work: every tutorial is built as a real project, tested against current versions, documented step by step, and revisited when the frameworks move on. A single in-depth tutorial routinely takes days — and that's before the bills behind it: servers, storage, build infrastructure, licenses, and the hardware everything is verified on.

If an article or tutorial here has saved you an afternoon of debugging, taught you a technique, or helped you ship, it created real value for you. Supporting it isn't charity — it's paying for professional work you found useful, at whatever price you choose. And if you can't, keep reading anyway. That's what the free part means.

## Ways to support

KO-FI

### Buy me a coffee

A one-time tip — no account, no subscription, any amount.

BOOKS & COURSES

### Buy the paid work

The paid tier of everything published free here — and you get something lasting in return.

SPREAD THE WORD

### Share an article

Send a tutorial to a colleague, cite an article, subscribe to the RSS feed.

Nothing here goes behind a paywall, whether you chip in or not. But if something on this site earned its keep today, this is where to say so.

**Support on Ko-fi —**