



New DevOps Download

How to protect your development environment from malicious and vulnerable components May 21|11AM EST

RS

COMPLIANCE / OPEN SOURCE

How Do Open Source Licenses Work? The Ultimate Guide

What are open source licenses and how do they work? Here's a guide for creators, users and others who want to take part in this wonderful open source movement.

Apr 12th, 2024 7:14am by [B. Cameron Gain](#)



Feature image by Jon Tyson on Unsplash



OXPOP

Try our new 5 second poll. It's fast. And it's fun!



New DevOps Download

How to protect your development environment from malicious and vulnerable components May 21|11AM EST

RS

I HAVE AN OPINION

We'd love to hear what you think.

Why Should I Care about the Open Source License?

You should care about an open source license whenever you engage with open source code, be it through usage, contribution or creation. The license set by the creator is crucial for respecting the intentions of both the creators and the contributor community, influencing aspects such as usage, commercialization, distribution and other project attributes.

That said, this Ultimate Guide is not intended to serve as legal advice or to be used as legal guidance for business plans or usage hopes for open source. Our intention here is to describe licenses for open source and what they mean for creators and users, extending to entrepreneurs, enthusiasts, researchers or anyone else who would like separate take part in this **open source movement**.

What Is an Open Source License?

An open source license is used to offer terms to respect the intention of the creator or creators and to determine usage of the **open source code** under the license that is allowed and restricted in some ways. For the user, it helps to offer guidance on how the code can be used, exploited or shared while respecting the license.

The **Open Source Initiative** (OSI) stipulates that a project must have a specific license that aligns with the OSI's definition to be considered open source. Licenses in compliance with **OSI's definition** of open source "allow software to be freely used, modified and shared." Over 100 different licenses exist that the **OSI**

A project when undergoing consideration for a particular license can go through

THE NEW STACK

The most popular licenses fall into different categories. Among the licenses that fall under the OSI's popular license categories and that have what the OSI deems a strong community include: [the Apache License](#), [the Common Development and Distribution License](#), [Eclipse Public License](#), various versions of the [GNU General Public License](#), [the Mozilla License](#), [the BSD License](#) and, of course, the popular and especially permissive [MIT License](#).

For the purposes of this explainer article, we will cover OSI licenses. However, in addition to OSI, there are other expert groups that offer listings and descriptions of software licenses that fall under the open source, free or free and open source. One such entity is the Debian project, which is a [Linux distribution developed by the Debian project](#). It offers both free and [open source software](#) as well as non-free software.

TRENDING STORIES

- [1. Valkey: A Redis Fork With a Future](#)
- [2. HashiCorp's Licensing Change is only the Latest Challenge to Open Source](#)
- [3. What's Next for Companies Built on Open Source?](#)
- [4. Linux Foundation Backs 'Valkey' Open Source Fork of Redis](#)
- [5. 'Open Source' Has a Definition, Let's Get Serious About Defending It](#)

Another project worth mentioning is the Fedora project, which contributes to the development of Fedora Linux, maintained by [Red Hat](#). It promotes a world-aware, open source, free, and open source software environment, built by a community of welcoming, open-minded individuals who support today's documentation.

The type of license a project creator chooses should be determined by legal counsel, as entering the realm of copyright law requires alignment with the project creator's intentions. The type of license, which only qualified legal

THE NEW STACK

utilize it. This stands in contrast to so-called "copyleft" licenses, like the GNU General Public License, which impose more stipulations. Under the OSI's definition of GNU GPL, rights are protected with two steps:

(1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions

"One of the beauties of open source software is the standardization of licenses approved by the Open Source Initiative as compliant with the Open Source Definition. This means that many people with a basic understanding of open source licenses don't need to go to legal counsel for advice but can generally rely on the established meaning of the license terms," Amanda Brock, CEO of OpenUK, told The New Stack. "This reliability and standardization is an important part of the free flow of open source software."

However, that is only the case if you understand the licenses, Brock said. "Knowing what copyleft and permissive licensing means is at the heart of this," Brock said. While there are around 80 approved licenses there are only a handful that are most used and knowing these and why people use them is generally enough.

How Do Open Source Licenses Compare to Free Software Licenses?

Open source can be equivalent to free software but in some cases not under certain definitions. Both are defined as such in that the code is open and free to use, but under the terms of a specific license.

Within the listings of the Free Software Foundation, it offers its own open source and free software licenses. The FSF's defines free software as:

Software that respects users' freedom and community. Roughly, it means that

However, there are some components or listings under the OSI's open source software licenses that the FSF (Free Software Foundation) does not consider to be free. Both free and open source software fall under the so-called FOSS (Free and Open Source Software) umbrella.

Is It Possible to Change the License?


The process of changing a license can be complicated and challenging, emphasizing the importance of selecting the right license initially. When altering an **open source project's** license, the new license must often comply with or be compatible with the original license, depending on its terms.

Ensuring that the changes align with the copyright holders' stakes in the project is crucial. This intricate process requires the guidance of competent legal counsel. It's advisable to establish proper licensing from the project's outset to avoid complexities later on.

What Are Some Noteworthy Examples of License Changes?

In August 2023, HashiCorp revealed it was changing the licensing terms of its leading Infrastructure as Code platform **Terraform**. It replaced its open source Mozilla Public License v2.0 (MPL 2.0) with the Business Source License (BSL) v1.1, which **restricts use of the code in production**.

This decision sparked controversy and an open source fork project emerged in response: the open source equivalent OpenTofu, an MPL-licensed forked version of Terraform's code **OpenTofu**, Supported by the Linux Foundation supports, OpenTofu 1.6 became **generally available** in January with added features compared to the original Terraform and which has received ample community support.

Notably, Grafana Labs has also adjusted its licenses for **open source projects** in the past. This move occurred in 2022 when Grafana decided to forgo its support  Cortex and launch an alternative to visualize metrics from Prometheus on a **time series database**.

the AGPLv3 license, Grafana Labs said.

HashiCorp's and Grafana's decisions to offer more restrictive licenses follows a trend that began to emerge in 2018 when **Redis Labs** and **MongoDB** opted to change their open licenses to restrict their open source product code from being used in production for commercial gain. Others that also have changed the licensing terms of their open source projects to become less permissive include **Confluent** and **Cockroach Labs**. They switched to the BSL, which makes the code open and free to use for non-commercial purposes and not in production.

Does Sharing Code on GitHub or GitLab Mean It's Protected by an Open Source License by Default?

The answer is no. In fact, the license helps to define and determine your **open source project**. While code can be shared or generally often shared on GitHub or GitLab, the creator of the code owns the code. It also does not belong to **Microsoft's GitHub** or **GitLab**.

As **GitHub states**, the creator risks further peril when other creators contribute and they are no longer able to use the code: "When you make a creative work (which includes code), the work is under exclusive copyright by default. Unless you include a license that specifies otherwise, nobody else can copy, distribute or modify your work without being at risk of take-downs, shake-downs or litigation. Once the work has other contributors (each a copyright holder), 'nobody' starts including you."

Different licenses offer varying degrees of **protection and freedom and users** should carefully review the stipulations, usually found in a LICENSE.txt, LICENSE.md or LICENSE.rst files or on in a README file on GitHub. Understanding the license helps determine how you can use, distribute, consume, copy or modify the code based on the specified restrictions.

When activating a license for an **open source project** on GitLab, the platform offers a user interface. Unlike with GitHub, the creator must sign in as a GitLab administrator. From the interface, the user chooses the type of license they would like to add with a checkbox to select during this process.

model, the license stipulations remain on platforms like GitHub. Your options with the forked code are determined by the original license model. A strict license might limit your usage or distribution but allows you to contribute changes back to the parent repository through suggested changes or pull requests.

With a permissive license like MIT, you can use the code from a fork to **create an entirely different open source project**, even with a different name and possibly a more restrictive license. The available options can vary depending on the license in place.

What Is the Best License on Which to Build a Business?

That is the million — or actually billion-dollar — question. Business strategies centered around open source vary, and there's no template for what makes a viable open source project commercially successful. For instance, initiating an **open source project as the core of a business** and then offering enterprise features on top has proven challenging, especially given limited venture capital funding and shorter runways to establish viable businesses today. So, the short answer is that it depends.

As Kubernetes expert **Kelsey Hightower** often emphasizes, there have been beautiful and robust open source projects resulting from significant efforts and investments by the founding company and the community. However, successfully monetizing the enterprise version remains a difficult bridge to cross.

In the realm of large open source projects, many successful ones weren't initially created to form the basis of a business model. Instead, they indirectly contributed to **business models or were used to solve technical problems**, attracting additional resources and input. For example, **Envoy**, a **service proxy started** by the ride-share company Lyft, became one of the most successful open source projects for Kubernetes. Kubernetes itself, launched by Google, formed the foundation for the cloud native movement, despite Google not being initially a cloud provider.

However, regardless of the project creators' goal for the open source project, success hinges on getting the license model right from the start.

THE NEW STACK

projects, like MongoDB, to offer a version as part of its cloud offering and charge for commercial support, turning the open source creator into a competitor for AWS customers. This led to disputes and tensions. Another instance involved Elasticsearch, which opted to make its code proprietary. In response, Amazon forked the project, turning it proprietary and offering paid services, prompting criticism by Elasticsearch.

Some observers, however, said AWS' move was necessary to counter Elasticsearch's commercial model for its use of these open source projects.

AWS is certainly legally able to take the code built from years of devotion and hard work devoted to open source projects while rebranding the open source tools and platforms and offering paid services to help use and manage the code (depending on the licenses). However, the cloud giant runs the risk of being perceived as betraying its customers and those who helped to create the projects, some observers say.

As mentioned above, AWS courted controversy in the open source community in 2019 when it began to offer MongoDB as part of its cloud offerings and to offer commercial support services for MongoDB. MongoDB, meanwhile, now offers commercial services based on its **open source stack** as mentioned above. Elastic opted to forgo Elasticsearch's and Kibana's Apache 2.0 licenses and replace those with an "Elastic License" and a Server Side Public License (SSPL), thus initiating a more commercial business model for Elasticsearch by Elastic.

In an email statement, Elastic wrote:

*"By incorporating all of our learnings from our experience and other companies who faced a similar decision, the introduction of the Elastic License v2 (ELv2) has provided much needed clarification within the market. We have since expanded our offerings — such as Elastic Cloud and other tools within the Elastic stack — and allow customers to deploy Elastic **products across their favorite public cloud or multi-cloud environments**. This strategic alignment has, in turn, fostered a stronger partnership between Elastic and cloud providers, like Amazon Web Services, Microsoft Azure and Google Cloud Platform, enabling our joint customers to benefit even more from our mutual investments in technology*

THE NEW STACK



cents at the local video arcade in the early 1980s. He then...

Read more from B. Cameron Gain →

Amazon Web Services, Cockroach Labs, Confluent, Microsoft, Red Hat and Redis are sponsors of The New Stack.

TNS owner Insight Partners is an investor in: Kubernetes.

SHARE THIS STORY



TRENDING STORIES

1. Valkey: A Redis Fork With a Future
2. HashiCorp's Licensing Change is only the Latest Challenge to Open Source
3. What's Next for Companies Built on Open Source?
4. Linux Foundation Backs 'Valkey' Open Source Fork of Redis
5. 'Open Source' Has a Definition, Let's Get Serious About Defending It

INSIGHTS FROM OUR SPONSORS



데일리샷, MongoDB Atlas로 스마트 주류 검색 서비스를 혁신하다

3 May 2024

MongoDB Provider for Entity Framework Core Now Generally Available

2 May 2024



Enhancing Network Resilience with Istio on Amazon EKS

6 May 2024

Flowpipe: A Cloud Scripting Engine for DevOps Workflows

30 April 2024

Lean Into Verified Software Development

8 April 2024



Automate Bot User creation and management with our new APIs

2 May 2024

Build your developer-defined "golden path" with the Kubernetes Gateway API and ngrok

25 April 2024

Reverse proxy vs. API gateway: Key differences explained

17 April 2024

TNS DAILY NEWSLETTER



Receive a free roundup of the

EMAIL ADDRESS

SUBSCRIBE

The New Stack does not sell your information or share it with unaffiliated third parties. By continuing, you agree to our [Terms of Use](#) and [Privacy Policy](#).

ARCHITECTURE

- Cloud Native Ecosystem
- Containers
- Edge Computing
- Microservices
- Networking
- Serverless
- Storage

ENGINEERING

- AI
- Large Language Models
- Frontend Development
- Software Development
- API Management
- Python
- JavaScript
- TypeScript
- WebAssembly
- Cloud Services
- Data
- Security

OPERATIONS

- Platform Engineering
- Operations
- CD
- h Careers
- Tech Culture

CHANNELS

- Podcasts
- Ebooks
- Events
- Newsletter
- TNS RSS Feeds



THE NEW STACK

- About / Contact
- Sponsors
- Sponsorship
- Contributions



Community created roadmaps, articles, resources and journeys for developers to help you choose your path and grow in your career.

- Frontend Developer Roadmap
- Backend Developer Roadmap
- Devops Roadmap

FOLLOW TNS

© The New Stack 2024

- Disclosures
- Terms of Use
- Advertising Terms & Conditions
- Privacy Policy
- Cookie Policy

