



## Contributing to VyOS

Daniil Baturin - 2020-09-07 - 0 Comments - in FAQ

### **Can I contribute to VyOS?**

Everyone is welcome to contribute to VyOS. Even if you are not a programmer, there are a lot of things you can do, including writing documentation, testing development builds and reporting issues.

### **What are the benefits for contributors?**

The goal of introducing LTS release image access subscriptions was to make VyOS development sustainable.

This is why we made them available for people who help the project move forward, either by purchasing a subscription and thus funding the work of the maintainers, or by participating in the project directly.

If you are contributing code, substantial amount of testing or documentation writing, or are an active VyOS evangelist, you are eligible for a free LTS image access subscription.

### **How do I contribute to documentation?**

VyOS documentation is now being developed in this repository:

<https://github.com/vyos/vyos-documentation/>

It's a Sphinx project that is automatically deployed to <https://docs.vyos.io>

The VyOS wiki (<http://wiki.vyos.net>) is going to be phased out when its content is migrated to the vyos-documentation project and this knowledge base.

### **How do I become a tester?**

Rolling release images are built nightly by our CI server, and can be found at

<https://downloads.vyos.io/?dir=rolling/current/amd64>

Issues should be reported to the [bug tracker](#).

## How do I become a developer?

VyOS source code is at [github.com/vyos](https://github.com/vyos)

The [vyos-build repository](#)/ contains the image build scripts.

All new features are now added to the [vyos-1x package](#) in an effort to consolidate the code and data.

Old packages inherited from Vyatta Core such as vyatta-cfg-system are considered legacy and are eliminated when the code they contain is redesigned and rewritten.

Since the release of VyOS 1.2.0 we no longer accept new features in Perl and shell, and no longer use the original command template system. As such, any such code will be rejected. All new code must follow the new [guidelines](#) created to ensure maintainability and enable us to introduce features formerly prevented by the limitations of the old config backend and old coding approach, such as parallelized commits, live rollbacks and so on.

Before you make a pull request, please create a task in Phabricator and reference it in your commit messages, e.g. "[BGP] T42: add support for overly large communities".