

# Using `tfenv` with OpenTofu

---

`tfenv` is a Terraform version manager. With the growing adoption of [OpenTofu](#)—the community-driven, open-source fork of Terraform—you can leverage `tfenv` to manage OpenTofu versions as well.

This guide explains:

- Installing `tfenv`
  - Using it with OpenTofu
  - Aliasing or symlinking `terraform` and `tf` to `tofu`
  - Recommendations on best practices
- 

## Prerequisites

- Git
  - A UNIX-like OS (Linux, macOS, or WSL)
  - Shell access (bash, zsh, etc.)
- 

## 1. Install `tfenv`

Option 1: Install via Homebrew (Recommended for macOS and Linuxbrew)

```
brew install tfenv
```

Make sure the `tfenv` path is in your shell config:

```
echo 'export PATH="$HOME/.tfenv/bin:$PATH"' >> ~/.bashrc # or ~/.zshrc
source ~/.bashrc # or ~/.zshrc
```

Option 2: Manual Installation

```
git clone https://github.com/tfutils/tfenv.git ~/.tfenv
echo 'export PATH="$HOME/.tfenv/bin:$PATH"' >> ~/.bashrc # or ~/.zshrc
source ~/.bashrc # or ~/.zshrc
```

Verify installation:

```
tfenv --version
```

## 2. Install OpenTofu via `tfenv`

Method 1: Manual install as a version

```
mkdir -p ~/.tfenv/versions/opentofu-latest
cd ~/.tfenv/versions/opentofu-latest
curl -LO
https://github.com/opentofu/opentofu/releases/latest/download/tofu_Linux_x86_64.zip # or use Darwin_arm64.zip for macOS
unzip tofu_*.zip
mv tofu terraform # tfenv expects the binary to be named 'terraform'
chmod +x terraform
```

Method 2: Symlink to existing tofu binary

```
mkdir -p ~/.tfenv/versions/opentofu-latest
ln -s "$(which tofu)" ~/.tfenv/versions/opentofu-latest/terraform
```

`tfenv` will treat this symlink as a valid version.

---

## 3. Set OpenTofu as active version

```
tfenv use opentofu-latest
terraform version
```

You should see output like:

```
OpenTofu v1.x.x
```

---

## 4. Alias or Symlink `terraform` and `tf` to `tofu`

Aliases

Add to your shell config (`~/.bashrc`, `~/.zshrc`):

```
alias terraform='tofu'
alias tf='tofu'
```

Reload your shell:

```
source ~/.bashrc # or ~/.zshrc
```

## Symlinks (optional)

Create system-wide or local symlinks if you prefer not to rely on shell aliases:

```
sudo ln -sf $(which tofu) /usr/local/bin/terraform  
sudo ln -sf $(which tofu) /usr/local/bin/tf
```

**Warning:** This can override existing Terraform installations. Only do this if you're committed to OpenTofu.

## 5. Set Global Default

```
tfenv global opentofu-latest
```

This ensures OpenTofu is used globally unless overridden by a `.terraform-version` file.

## Recommendation: Alias vs Symlink

Method	Pros	Cons
Alias	Easy to add/remove, session-scoped	Doesn't work in scripts unless exported
Symlink	Works globally, including in scripts	Can be destructive if you still use Terraform

### Best Practice:

Use **aliases** during transition or testing. Use **symlinks** only when you're fully committed to OpenTofu and want full CLI compatibility.

## Conclusion

You can seamlessly use OpenTofu with `tfenv` and migrate away from Terraform with minimal disruption. Choose the method that best fits your workflow and risk tolerance.