FLOX

Nixifying your provisioning problems with Nix & OpenTofu

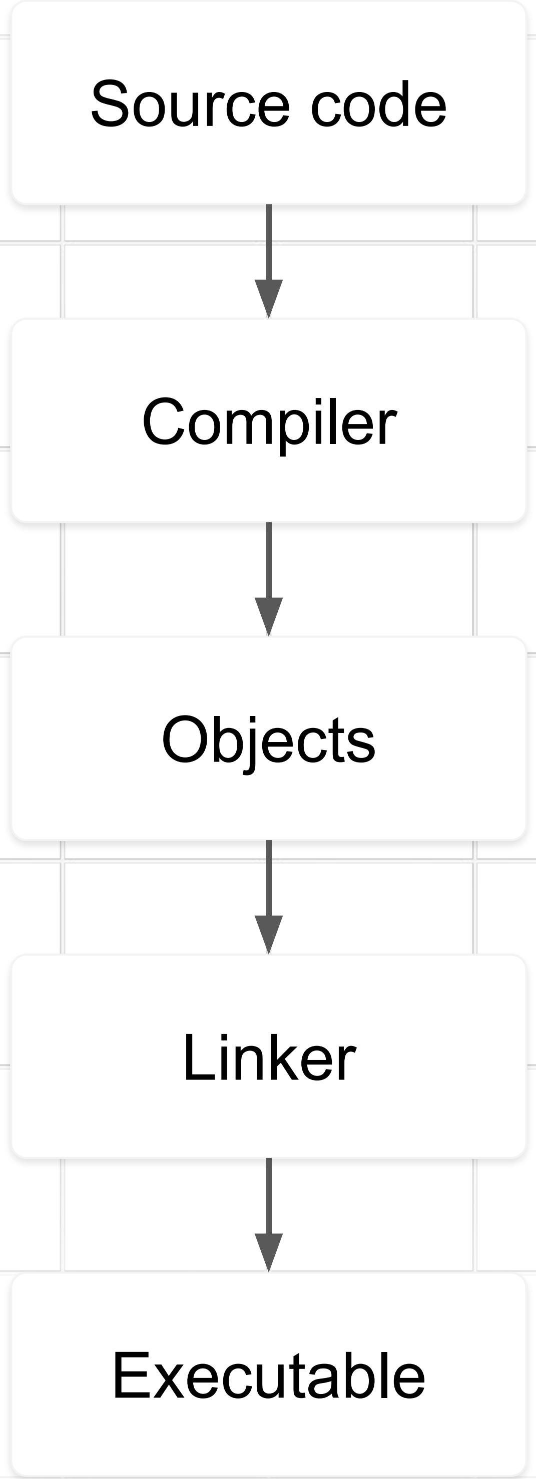
Bryan HONOF

flox.dev

Let's start with an idea

FLOX

A tale as old as time



Source code

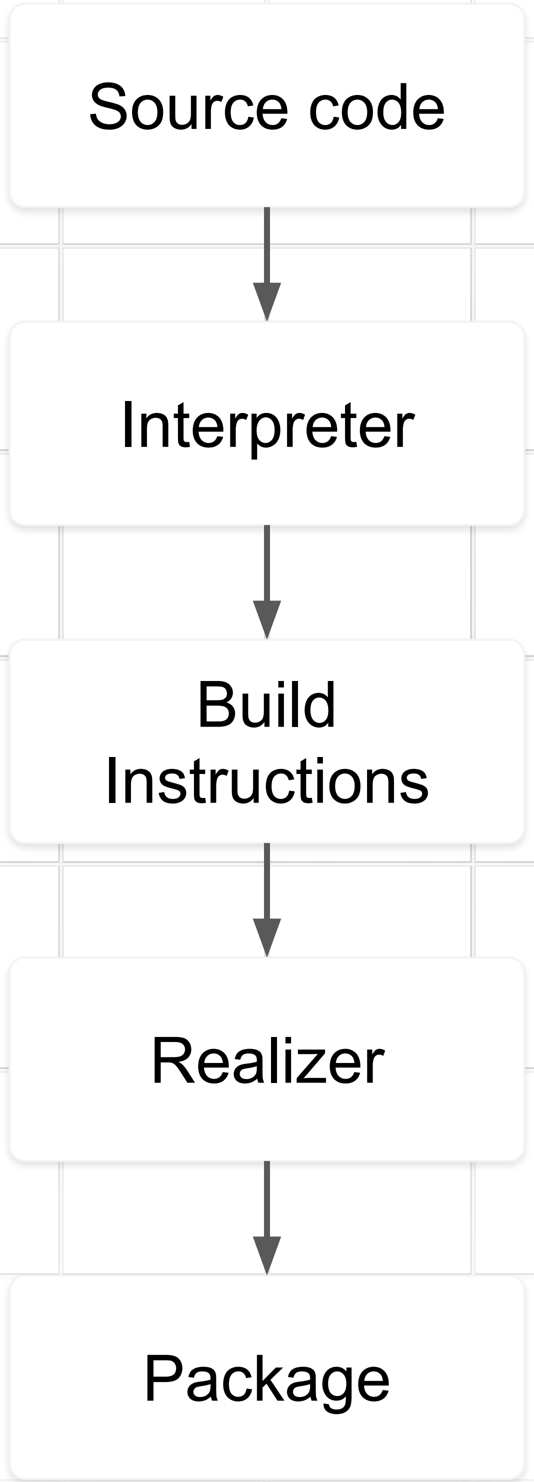
Compiler

Objects

Linker

Executable

What if…


Source code

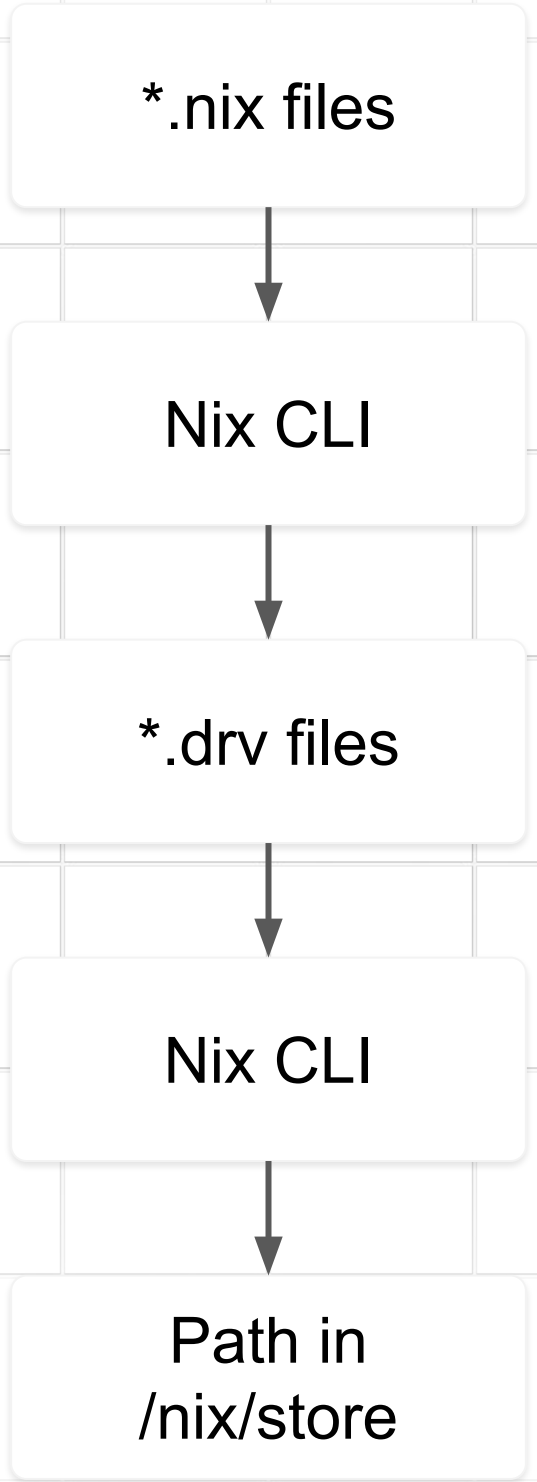
Interpreter

Build Instructions

Realizer

Package

It already exists!



So, what is Nix?

Well…

The term "Nix" is overloaded

- Nix, as in the CLI tool you interface with
- Nix, as in the language you write into *.nix files
- Nix, as in nixpkgs the package repository
- Nix, as in NixOS the Linux distribution

And a bunch of community projects having Nix somewhere in their name

The term "Nix" is overloaded

- Nix, as in the CLI tool you interface with
- Nix, as in the language you write into *.nix files
- Nix, as in nixpkgs the package repository
- Nix, as in NixOS the Linux distribution

And a bunch of community projects having Nix somewhere in their name

Great, another package manager 🙄

FLOX

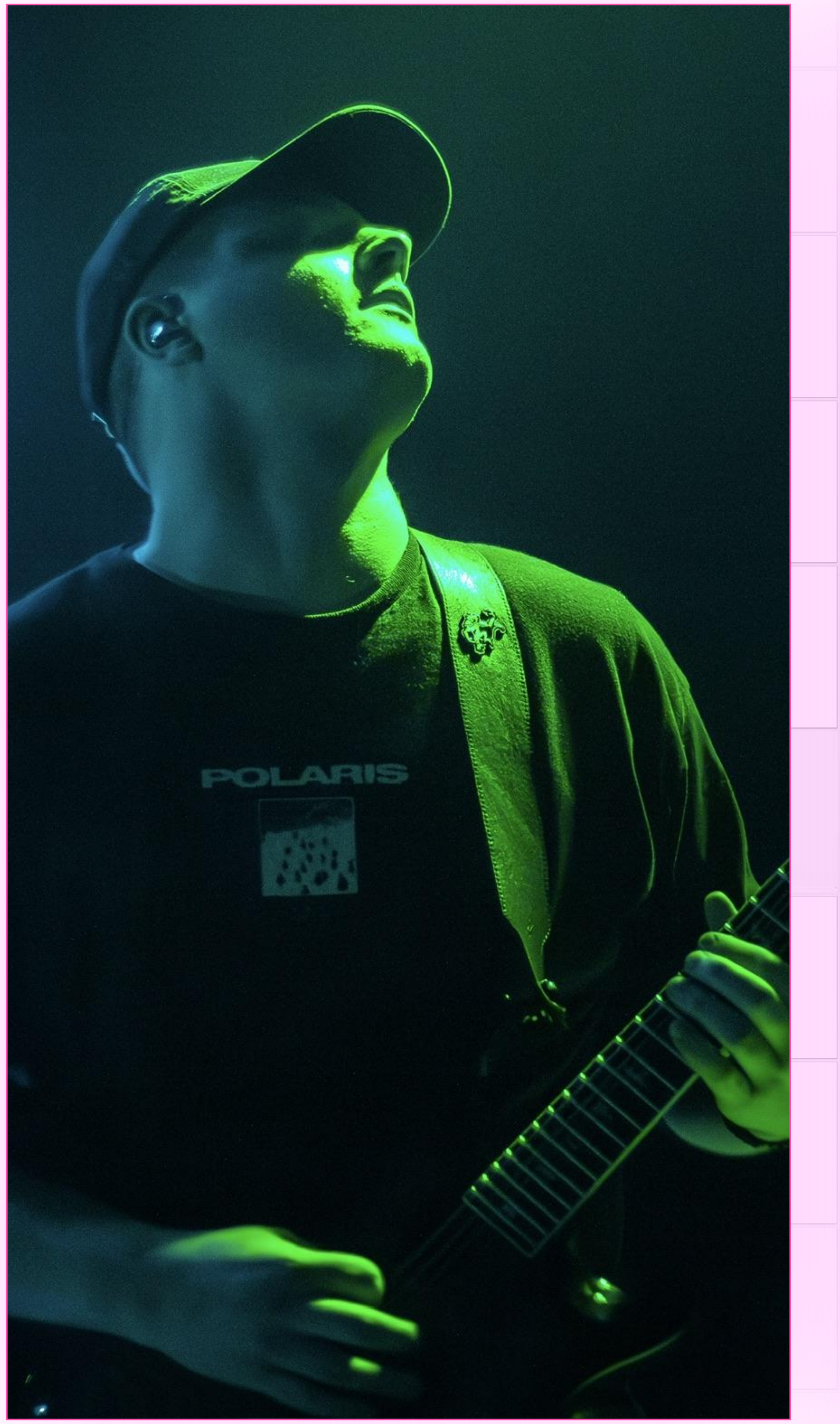
Cool, another package manager 🙄
With superpowers!

```
$ # What Python do I currently have?
$ python --version
zsh: command not found: python
$ # Oh...
$ # Let me hop in a shell with Python
$ nix shell --impure nixpkgs#python2
$ python --version
Python 2.7.18.8
$ # Wait, didn't Python 2 get deprecated?
$ exit
$ nix shell nixpkgs#python3
$ python --version
Python 3.11.8
$ # Back to my clean shell we go
$ exit
$ python --version
zsh: command not found: python
$ # I also don't need these packages anymore
$ nix store gc
```

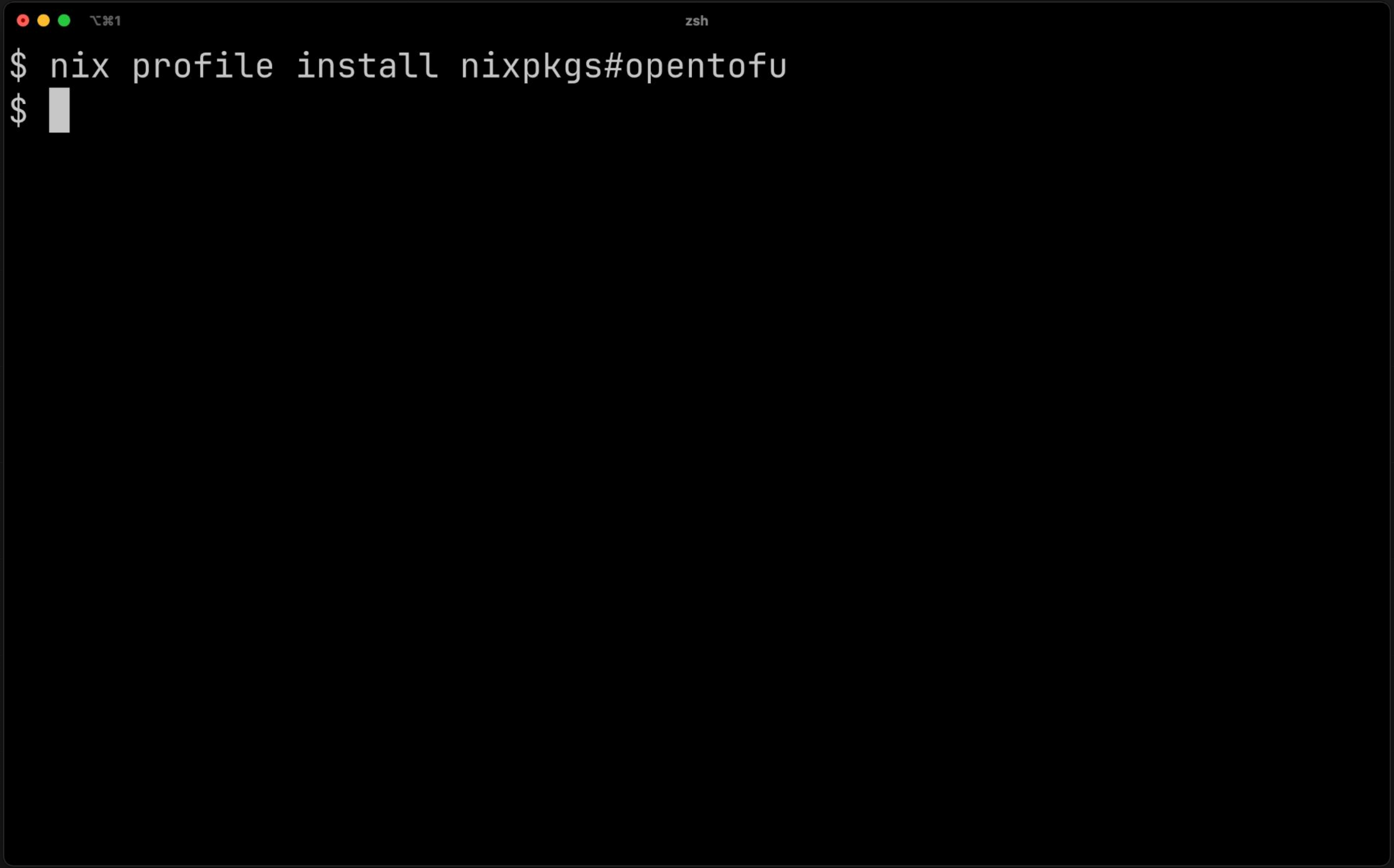
Wait, multiple versions of the same software?

FLOX

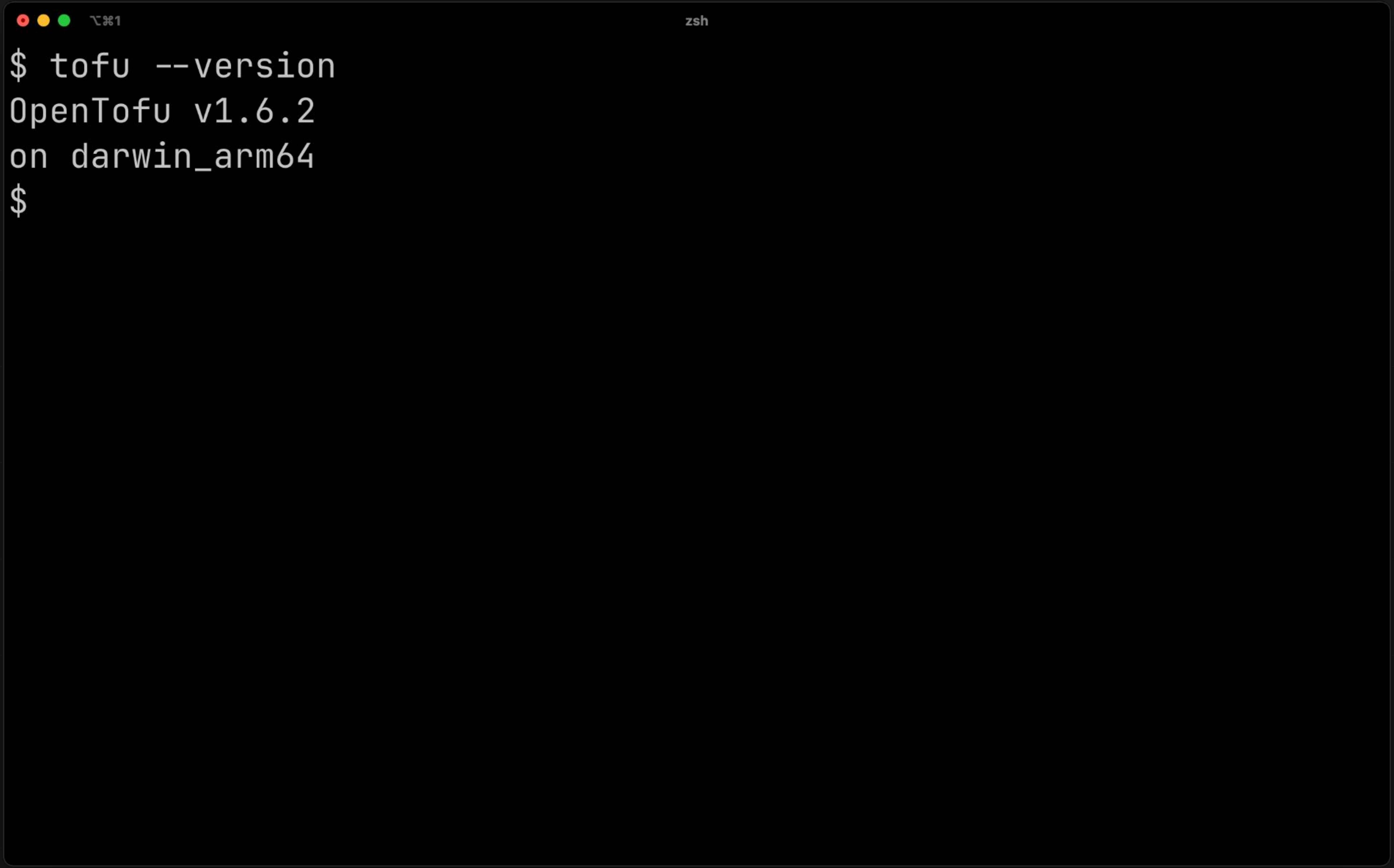
Let's do some software spelunking



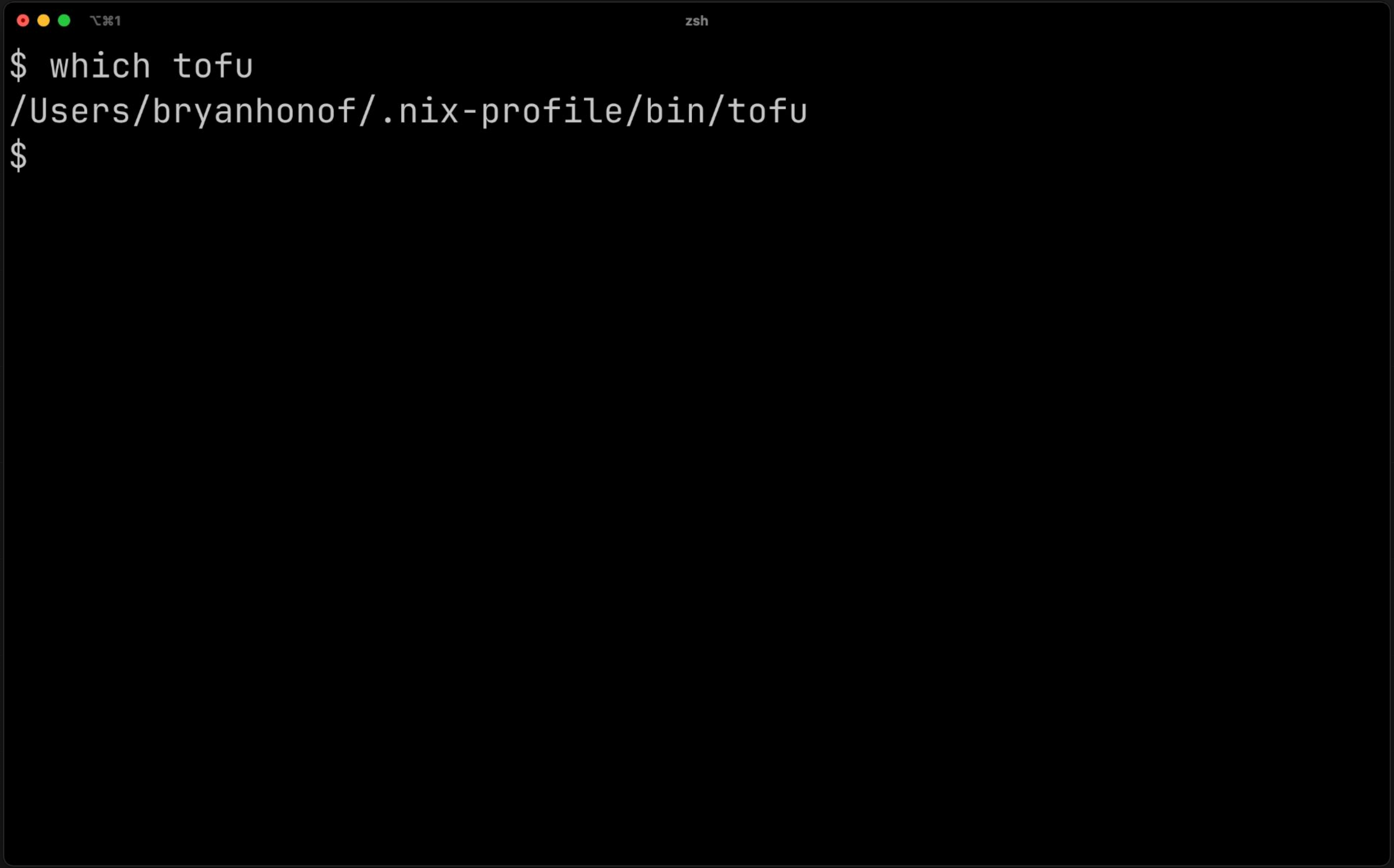
```
$ whoami
name:    Bryan HONOF
role:    Software Engineer @ Flox
email:   bryan@flox.dev
www:     www.bryanhonof.be
socials:
  - 🐘 @bryanhonof@mastodon.social
  - 🐦 @HonofBryan
  - 🧳 in/bryanhonof
```



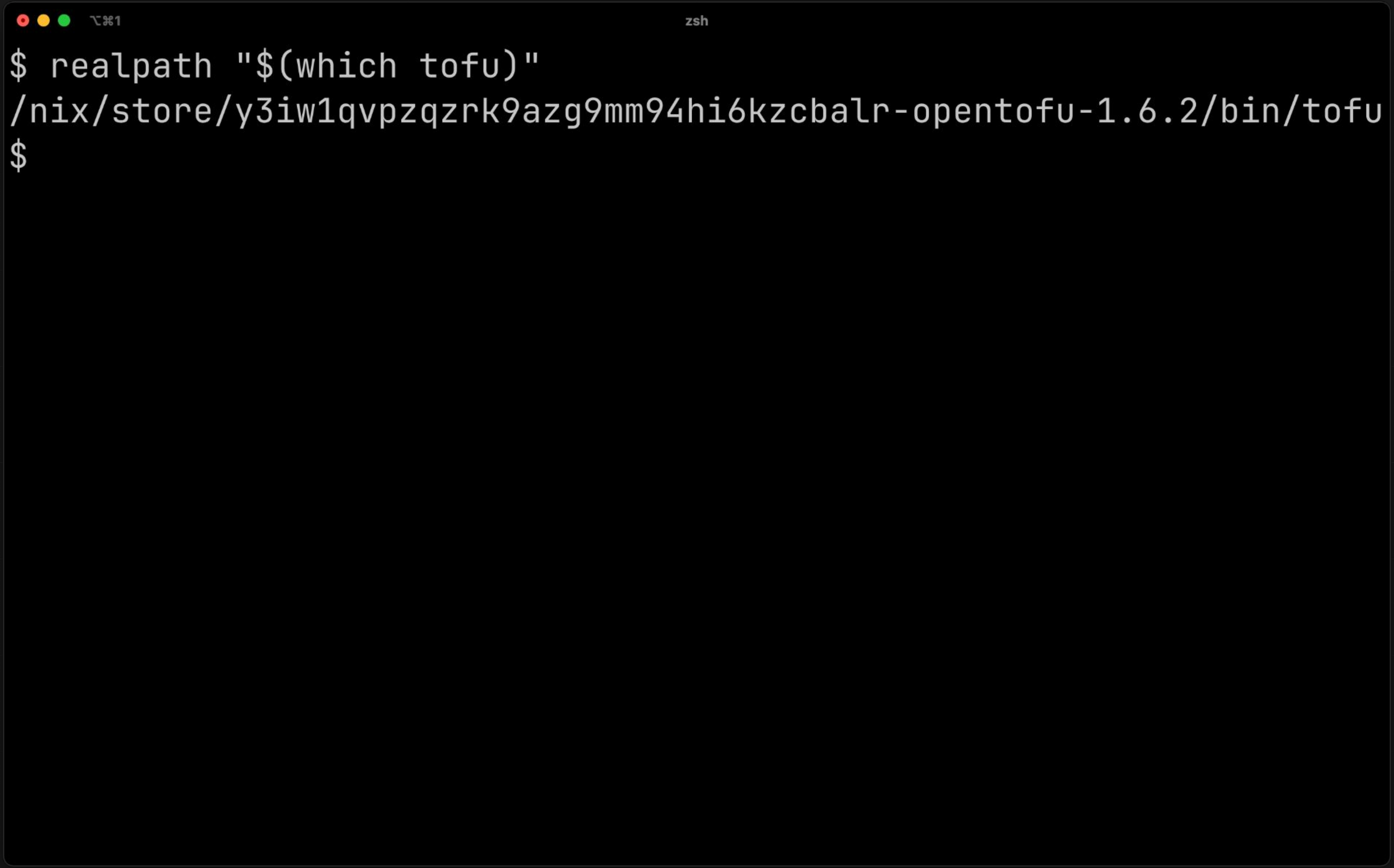
```
$ nix profile install nixpkgs#opentofu
$ 
```



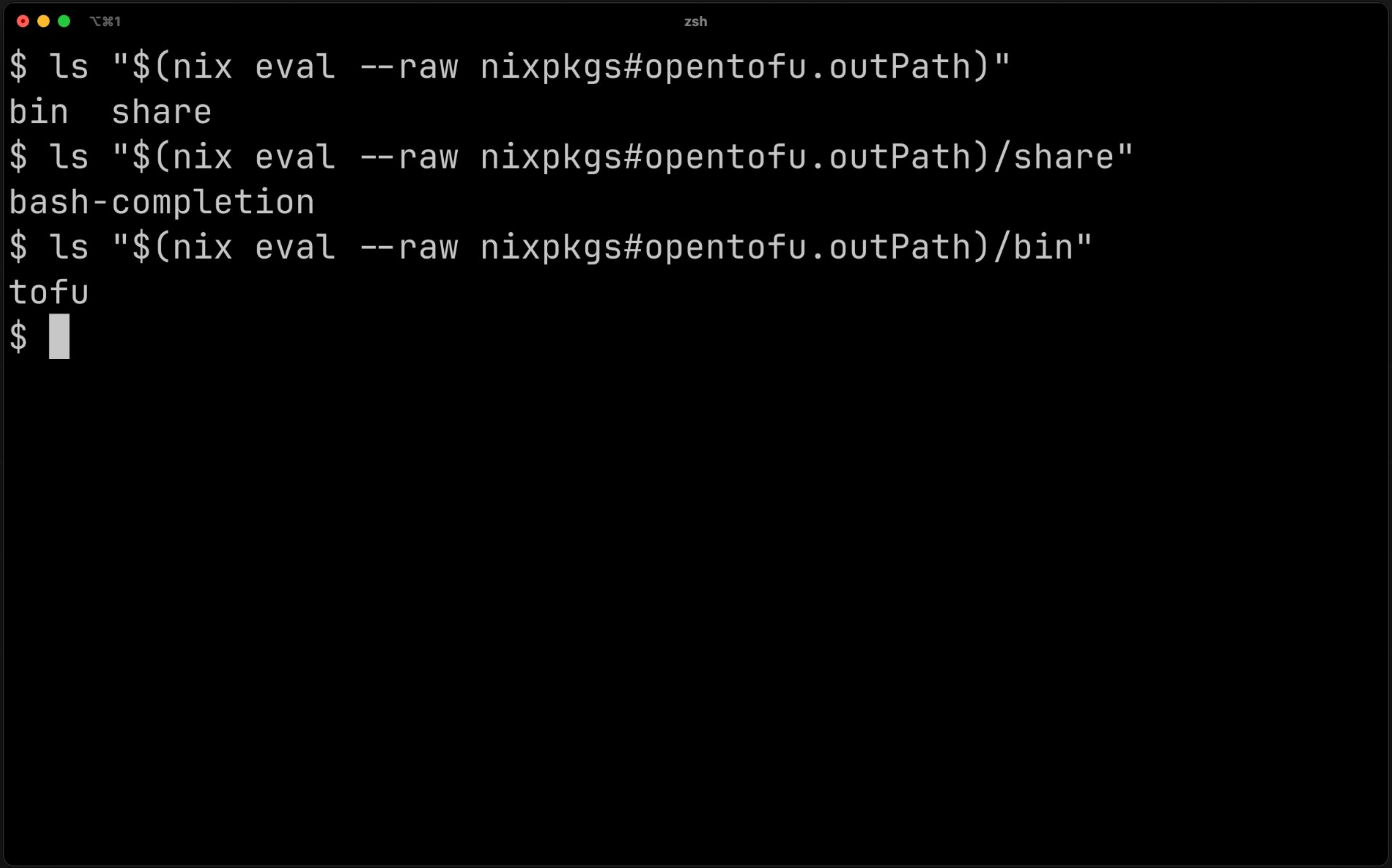
```
$ tofu --version
OpenTofu v1.6.2
on darwin_arm64
$
```



```
$ which tofu
/Users/bryanhonof/.nix-profile/bin/tofu
$
```

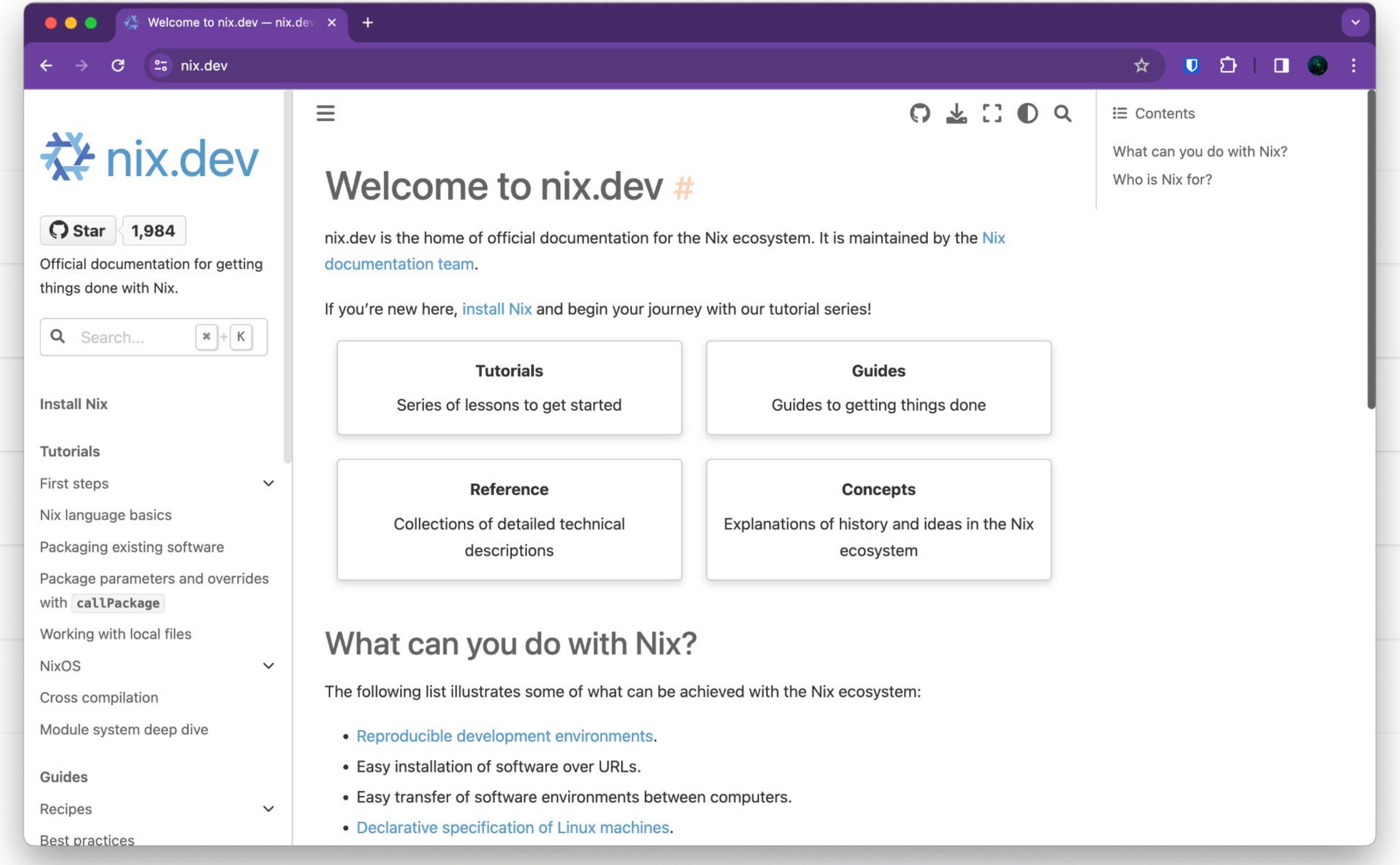


```
$ realpath "$(which tofu)"
/nix/store/y3iw1qvpzqzrk9azg9mm94hi6kzcbalr-opentofu-1.6.2/bin/tofu
$
```



```
$ ls "$(nix eval --raw nixpkgs#opentofu.outPath)"
bin  share
$ ls "$(nix eval --raw nixpkgs#opentofu.outPath)/share"
bash-completion
$ ls "$(nix eval --raw nixpkgs#opentofu.outPath)/bin"
tofu
$ 
```

Let me demo this live!



Welcome to nix.dev — nix.dev

nix.dev

nix.dev

Star 1,984

Official documentation for getting things done with Nix.

Search... ⌘ + K

Install Nix

Tutorials

First steps

Nix language basics

Packaging existing software

Package parameters and overrides with `callPackage`

Working with local files

NixOS

Cross compilation

Module system deep dive

Guides

Recipes

Best practices

Welcome to nix.dev

nix.dev is the home of official documentation for the Nix ecosystem. It is maintained by the Nix documentation team.

If you're new here, install Nix and begin your journey with our tutorial series!

Tutorials
Series of lessons to get started

Guides
Guides to getting things done

Reference
Collections of detailed technical descriptions

Concepts
Explanations of history and ideas in the Nix ecosystem

What can you do with Nix?

The following list illustrates some of what can be achieved with the Nix ecosystem:

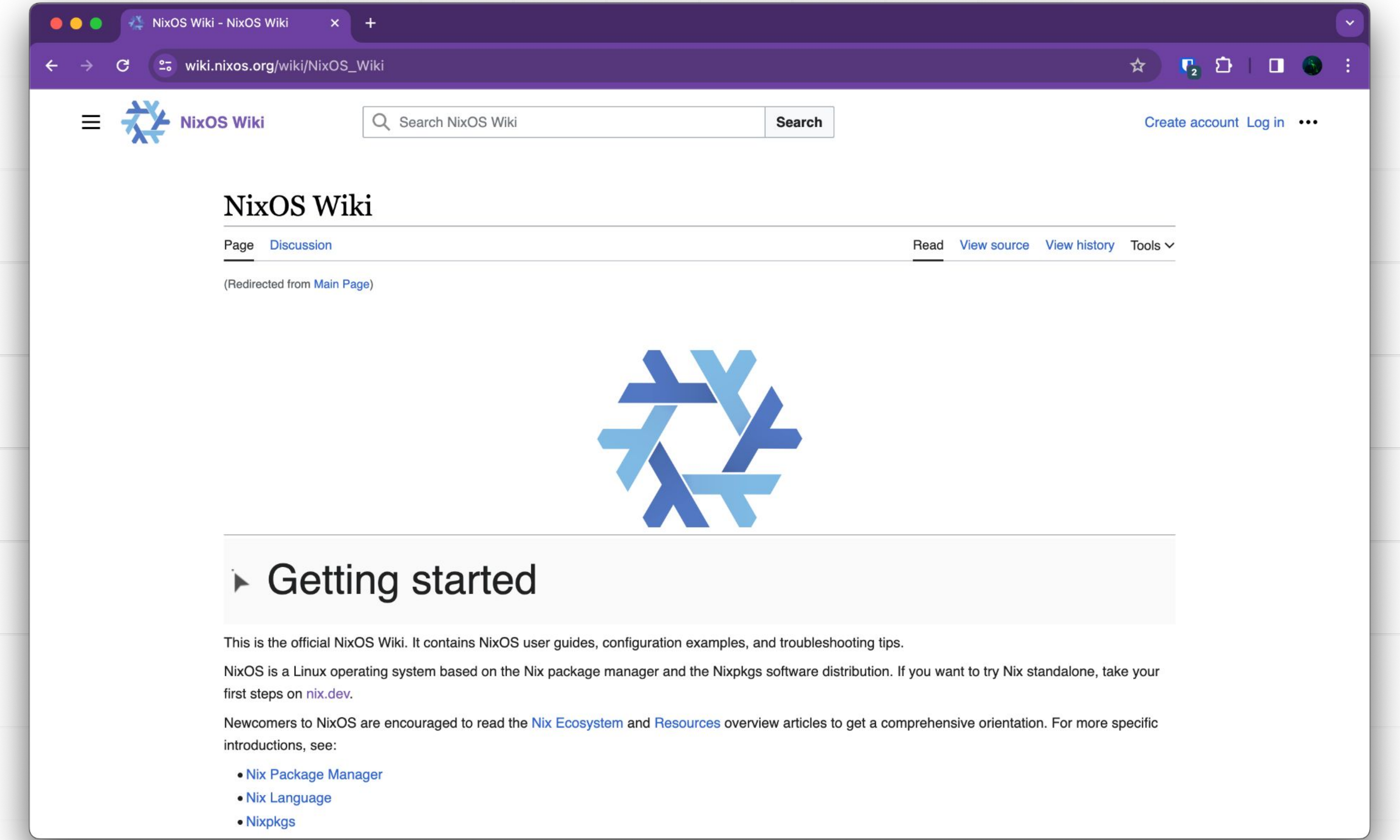
- Reproducible development environments.
- Easy installation of software over URLs.
- Easy transfer of software environments between computers.
- Declarative specification of Linux machines.

Contents

What can you do with Nix?

Who is Nix for?

FLOX



NixOS Wiki

(Redirected from Main Page)

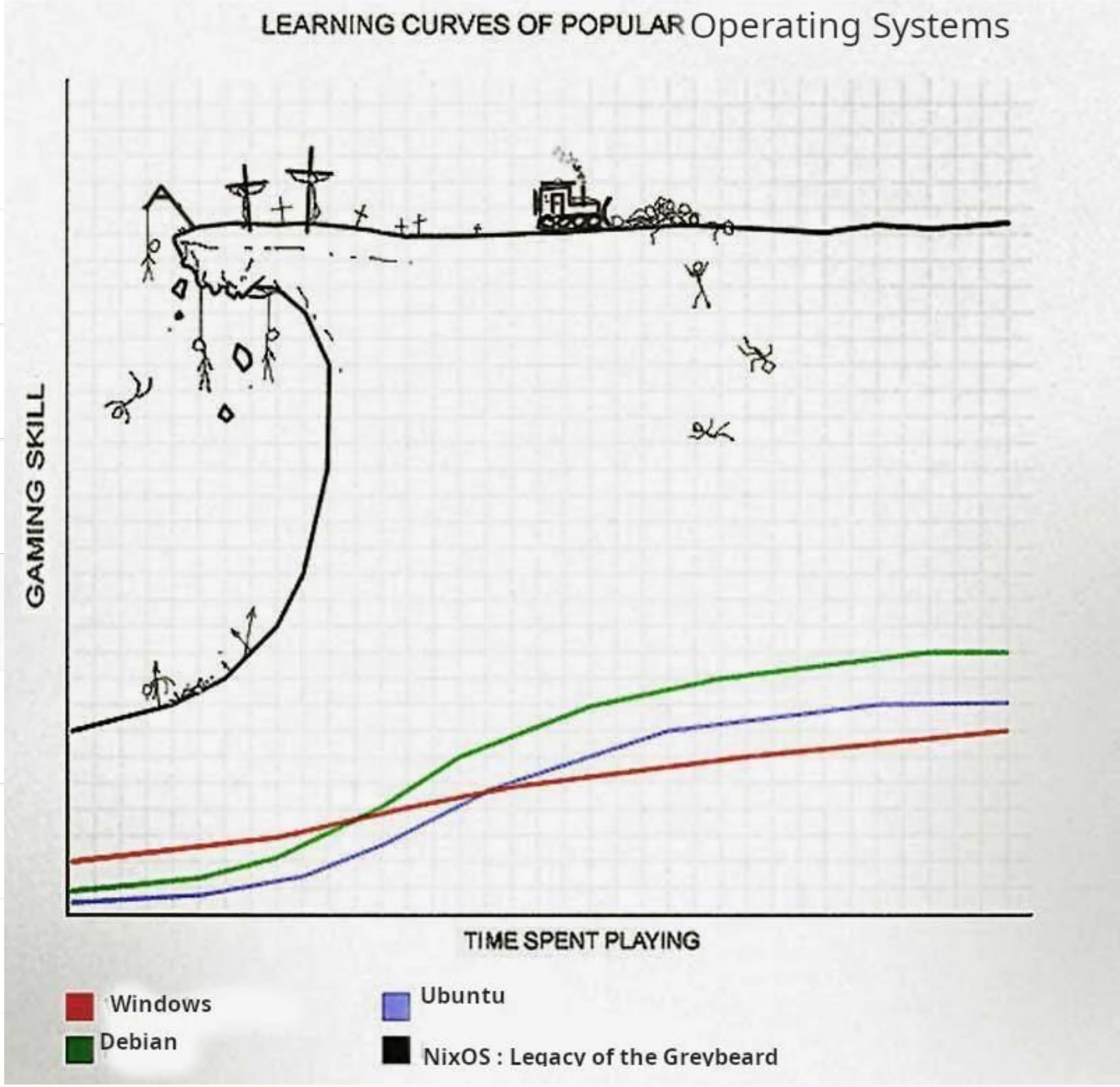
Getting started

This is the official NixOS Wiki. It contains NixOS user guides, configuration examples, and troubleshooting tips.

NixOS is a Linux operating system based on the Nix package manager and the Nixpkgs software distribution. If you want to try Nix standalone, take your first steps on nix.dev.

Newcomers to NixOS are encouraged to read the Nix Ecosystem and Resources overview articles to get a comprehensive orientation. For more specific introductions, see:

- Nix Package Manager
- Nix Language
- Nixpkgs


LEARNING CURVES OF POPULAR Operating Systems
GAMING SKILL
TIME SPENT PLAYING
🟥 Windows
🟩 Debian
🟦 Ubuntu
⬛ NixOS : Legacy of the Greybeard

FLOX

flox.dev