



Oh My Zsh is an open source, community-driven framework for managing your [zsh](#) configuration.

Sounds boring. Let's try again.

Oh My Zsh will not make you a 10x developer...but you may feel like one.

Once installed, your terminal shell will become the talk of the town *or your money back!* With each keystroke in your command prompt, you'll take advantage of the hundreds of powerful plugins and beautiful themes. Strangers will come up to you in cafés and ask you, "*that is amazing! are you some sort of genius?*"

Finally, you'll begin to get the sort of attention that you have always felt you deserved. ...or maybe you'll use the time that you're saving to start flossing more often. 😊

To learn more, visit [ohmyz.sh](#), follow [@ohmyzsh](#) on Twitter, and join us on [Discord](#).

 CI **passing**  Follow [@ohmyzsh](#) 48k  chat 1059 online  Gitpod **ready**

 security bounty **up to \$750 + CVE**

► [Table of Contents](#)

Getting Started

Prerequisites

- A Unix-like operating system: macOS, Linux, BSD. On Windows: WSL2 is preferred, but cygwin or msys also mostly work.
- [Zsh](#) should be installed (v4.3.9 or more recent is fine but we prefer 5.0.8 and newer). If not pre-installed (run `zsh --version` to confirm), check the following wiki instructions here: [Installing ZSH](#)
- `curl` or `wget` should be installed
- `git` should be installed (recommended v2.4.11 or higher)

Basic Installation

Oh My Zsh is installed by running one of the following commands in your terminal. You can install this via the command-line with either `curl`, `wget` or another similar tool.

Method	Command
curl	<pre>sh -c "\$(curl -fsSL https://raw.githubusercontent.com/ohmyzsh/ohmyzsh/master/tools/install.sh)"</pre>
wget	<pre>sh -c "\$(wget -O- https://raw.githubusercontent.com/ohmyzsh/ohmyzsh/master/tools/install.sh)"</pre>
fetch	<pre>sh -c "\$(fetch -o - https://raw.githubusercontent.com/ohmyzsh/ohmyzsh/master/tools/install.sh)"</pre>

Note that previous configuration in `.zshrc` will be placed in `.pre-oh-my-zsh`. After installation you can move configuration you want to preserve into `.zshrc`.

Manual inspection

It's a good idea to inspect the install script from projects you don't yet know. You can do that by downloading the install script first, looking through it so everything looks normal, then running it:

```
wget https://raw.githubusercontent.com/ohmyzsh/ohmyzsh/master/tools/install.sh
sh install.sh
```

Using Oh My Zsh

Plugins

Oh My Zsh comes with a shitload of plugins for you to take advantage of. You can take a look in the [plugins](#) directory and/or the [wiki](#) to see what's currently available.

Enabling Plugins

Once you spot a plugin (or several) that you'd like to use with Oh My Zsh, you'll need to enable them in the `.zshrc` file. You'll find the `zshrc` file in your `$HOME` directory. Open it with your favorite text editor and you'll see a spot to list all the plugins you want to load.

```
vi ~/.zshrc
```

For example, this might begin to look like this:

```
plugins=(
  git
  bundler
  dotenv
  macos
  rake
  rbenv
```

```
ruby
)
```

*Note that the plugins are separated by whitespace (spaces, tabs, new lines...). **Do not** use commas between them or it will break.*

Using Plugins

Each plugin includes a **README**, documenting it. This README should show the aliases (if the plugin adds any) and extra goodies that are included in that particular plugin.

Themes

We'll admit it. Early in the Oh My Zsh world, we may have gotten a bit too theme happy. We have over one hundred and fifty themes now bundled. Most of them have [screenshots](#) on the wiki (We are working on updating this!). Check them out!

Selecting a Theme

Robby's theme is the default one. It's not the fanciest one. It's not the simplest one. It's just the right one (for him).

Once you find a theme that you'd like to use, you will need to edit the `~/.zshrc` file. You'll see an environment variable (all caps) in there that looks like:

```
ZSH_THEME="robbyrussell"
```

To use a different theme, simply change the value to match the name of your desired theme. For example:

```
ZSH_THEME="agnoster" # (this is one of the fancy ones)
# see https://github.com/ohmyzsh/ohmyzsh/wiki/Themes#agnoster
```

Note: many themes require installing the [Powerline Fonts](#) in order to render properly.

Open up a new terminal window and your prompt should look something like this:

```
~> cd testproject
~/testproject ~ master gco detached-head-state -q
~/testproject ~ fdffaf6 touch dirty-working-directory
~/testproject ~ fdffaf6± cd
~> ssh milly
Welcome to Ubuntu 11.04 (GNU/Linux 2.6.18-308.8.2.el5.028stab101.1 x86_64)
Last login: Wed Sep 26 03:42:49 2012 from 71-215-222-90.mpls.qwest.net
agnoster@milly ~
Connection to milly.agnoster.net closed.
~> sudo -s
Password:
~ root@Arya ~ top &
[1] 34523
[1] + 34523 suspended (tty output) top
~ root@Arya ~ rm no-such-file
rm: no-such-file: No such file or directory
~ root@Arya ~ kill %%
[1] + 34523 terminated top
~ root@Arya ~
~>
```

In case you did not find a suitable theme for your needs, please have a look at the wiki for [more of them](#).

If you're feeling feisty, you can let the computer select one randomly for you each time you open a new terminal window.

```
ZSH_THEME="random" # (...please let it be pie... please be some pie..)
```

And if you want to pick random theme from a list of your favorite themes:

```
ZSH_THEME_RANDOM_CANDIDATES=(
  "robbyrussell"
  "agnoster"
)
```

If you only know which themes you don't like, you can add them similarly to an ignored list:

```
ZSH_THEME_RANDOM_IGNORED=(pygmalion tjkirch_mod)
```

FAQ

If you have some more questions or issues, you might find a solution in our [FAQ](#).

Advanced Topics

If you're the type that likes to get their hands dirty, these sections might resonate.

Advanced Installation

Some users may want to manually install Oh My Zsh, or change the default path or other settings that the installer accepts (these settings are also documented at the top of the install script).

Custom Directory

The default location is `~/.oh-my-zsh` (hidden in your home directory, you can access it with `cd ~/.oh-my-zsh`)

If you'd like to change the install directory with the `ZSH` environment variable, either by running `export ZSH=/your/path` before installing, or by setting it before the end of the install pipeline like this:

```
ZSH="$HOME/.dotfiles/oh-my-zsh" sh install.sh
```

Unattended install

If you're running the Oh My Zsh install script as part of an automated install, you can pass the flag `--unattended` to the `install.sh` script. This will have the effect of not trying to change the default shell, and also won't run `zsh` when the installation has finished.

```
sh -c "$(curl -fsSL https://raw.githubusercontent.com/ohmyzsh/ohmyzsh/master/tools/install.sh) " "" --unattended
```

Installing from a forked repository

The install script also accepts these variables to allow installation of a different repository:

- `REPO` (default: `ohmyzsh/ohmyzsh`): this takes the form of `owner/repository` . If you set this variable, the installer will look for a repository at `https://github.com/{owner}/{repository}` .
- `REMOTE` (default: `https://github.com/${REPO}.git`): this is the full URL of the git repository clone. You can use this setting if you want to install from a fork that is not on GitHub (GitLab, Bitbucket...) or if you want to clone with SSH instead of HTTPS (`git@github.com:user/project.git`).

NOTE: it's incompatible with setting the `REPO` variable. This setting will take precedence.

- `BRANCH` (default: `master`): you can use this setting if you want to change the default branch to be checked out when cloning the repository. This might be useful for testing a Pull Request, or if you want to use a branch other than `master` .

For example:

```
REPO=apjanke/oh-my-zsh BRANCH=edge sh install.sh
```

Manual Installation

1. Clone the repository

```
git clone https://github.com/ohmyzsh/ohmyzsh.git ~/.oh-my-zsh
```

2. Optionally, backup your existing `~/.zshrc` file

```
cp ~/.zshrc ~/.zshrc.orig
```

3. Create a new zsh configuration file

You can create a new zsh config file by copying the template that we have included for you.

```
cp ~/.oh-my-zsh/templates/zshrc.zsh-template ~/.zshrc
```

4. Change your default shell

```
chsh -s $(which zsh)
```

You must log out from your user session and log back in to see this change.

5. Initialize your new zsh configuration

Once you open up a new terminal window, it should load zsh with Oh My Zsh's configuration.

Installation Problems

If you have any hiccups installing, here are a few common fixes.

- You *might* need to modify your `PATH` in `~/.zshrc` if you're not able to find some commands after switching to `oh-my-zsh`.
- If you installed manually or changed the install location, check the `ZSH` environment variable in `~/.zshrc`.

Custom Plugins and Themes

If you want to override any of the default behaviors, just add a new file (ending in `.zsh`) in the `custom/` directory.

If you have many functions that go well together, you can put them as a `XYZ.plugin.zsh` file in the `custom/plugins/` directory and then enable this plugin.

If you would like to override the functionality of a plugin distributed with Oh My Zsh, create a plugin of the same name in the `custom/plugins/` directory and it will be loaded instead of the one in `plugins/`.

Getting Updates

By default, you will be prompted to check for updates every 2 weeks. You can choose other update modes by adding a line to your `~/.zshrc` file, **before Oh My Zsh is loaded**:

1. Automatic update without confirmation prompt:

```
zstyle ':omz:update' mode auto
```

2. Just offer a reminder every few days, if there are updates available:

```
zstyle ':omz:update' mode reminder
```

3. To disable automatic updates entirely:

```
zstyle ':omz:update' mode disabled
```

NOTE: you can control how often Oh My Zsh checks for updates with the following setting:

```
# This will check for updates every 7 days
zstyle ':omz:update' frequency 7
# This will check for updates every time you open the terminal (not recommended)
zstyle ':omz:update' frequency 0
```

Manual Updates

If you'd like to update at any point in time (maybe someone just released a new plugin and you don't want to wait a week?) you just need to run:

```
omz update
```

Magic! 🎩

Uninstalling Oh My Zsh

Oh My Zsh isn't for everyone. We'll miss you, but we want to make this an easy breakup.

If you want to uninstall `oh-my-zsh`, just run `uninstall_oh_my_zsh` from the command-line. It will remove itself and revert your previous `bash` or `zsh` configuration.

How do I contribute to Oh My Zsh?

Before you participate in our delightful community, please read the [code of conduct](#).

I'm far from being a [Zsh](#) expert and suspect there are many ways to improve – if you have ideas on how to make the configuration easier to maintain (and faster), don't hesitate to fork and send pull requests!

We also need people to test out pull-requests. So take a look through [the open issues](#) and help where you can.

See [Contributing](#) for more details.

Do NOT send us themes

We have (more than) enough themes for the time being. Please add your theme to the [external themes](#) wiki page.

Contributors

Oh My Zsh has a vibrant community of happy users and delightful contributors. Without all the time and help from our contributors, it wouldn't be so awesome.

Thank you so much!

Follow Us

We're on social media:

- [@ohmyzsh](#) on Twitter. You should follow it.
- [Facebook](#) poke us.
- [Instagram](#) tag us in your post showing Oh My Zsh!

- [Discord](#) to chat with us!

Merchandise

We have [stickers, shirts, and coffee mugs available](#) for you to show off your love of Oh My Zsh. Again, you will become the talk of the town!

License

Oh My Zsh is released under the [MIT license](#).

About Planet Argon



Oh My Zsh was started by the team at [Planet Argon](#), a [Ruby on Rails development agency](#). Check out our [other open source projects](#).