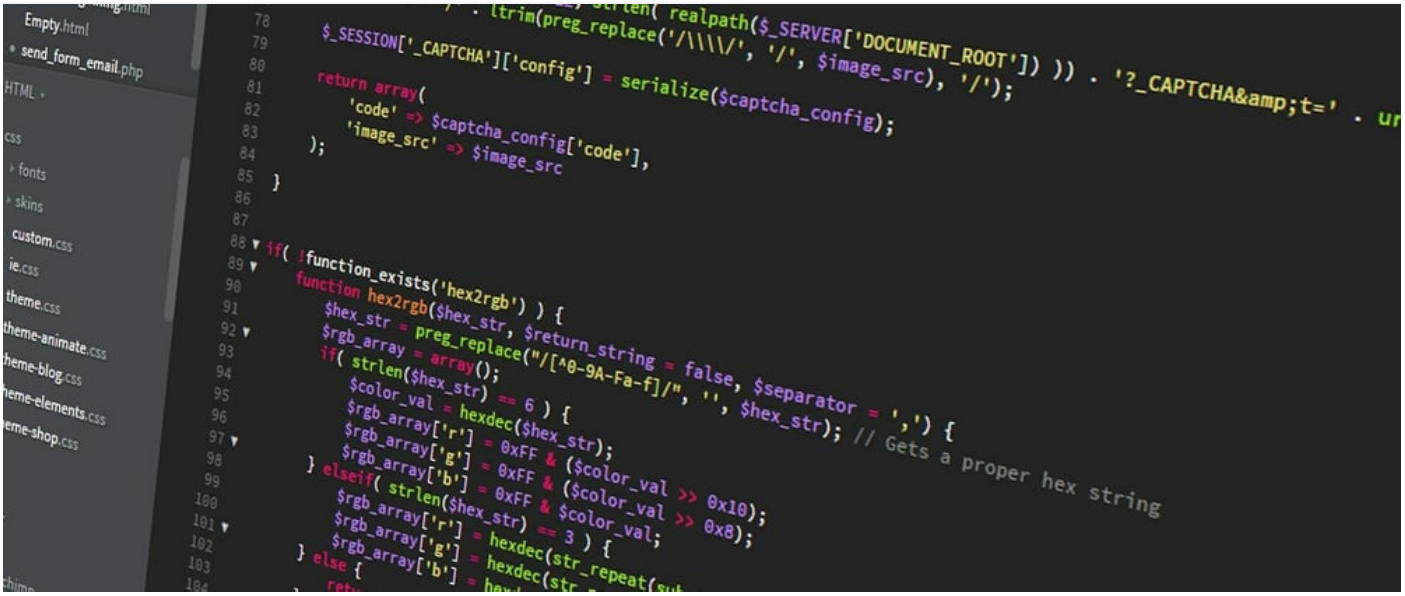


Dominik Zarsky
Posted on Mar 2, 2021 • Originally published at blog.dzarsky.eu

💖 15 🦄 2

How To Backup Your VS Code Extensions And Settings

#vscode #backup #bash #linux

Backup your Visual Studio Code extension list and configuration files using Linux terminal.

Having the same working environment across multiple machines is always a good thing. This might be a case for your system configuration files (dotfiles) and your development tools alike. What I'm going to show you is a simple script that can help with the task of backing up your Visual Studio Code settings and extensions for use on your secondary PC, virtual environment or even your computer at work.

Setting up the backup

For this article, I'm going to suppose that you have VS Code already installed on your system and want to backup the settings and extensions you've spent some of your precious time setting up. We'll be using the old trusty Linux terminal magic and VS Code's command line utility *code*.

Our basic command for exporting the list of extensions is going to look like this:

```
code --list-extensions | xargs -L 1 echo code --install-extension
```

The *code --list-extensions* command exports the list of installed VS Code extensions which we then pipe into xargs, this transforms each extension's name into an extension install command. The output of this command looks like this:

```
code --install-extension anseki.vscode-color
code --install-extension autsenc.laravel-blade-spacer
code --install-extension Dart-Code.flutter
```

As you see we've successfully transformed the extension names to installation commands via xargs. You can output these into a file like this:

```
mkdir /home/$USER/code-backup && cd /home/$USER/code-backup
code --list-extensions | xargs -L 1 echo code --install-extension > ext
```

Now we also need to export the configuration files. These usually reside in your .config directory. The config files are saved in JSON format, so to back these up, you would simply tar them and save them somewhere in the cloud. Alternatively, you could push these into a Git repo.

```
cd /home/$USER/.config/Code/User
tar -czf code-config.tar.gz *.json
mv code-config.tar.gz /home/$USER/code-backup
```

In your code-backup folder, you should have your config files in a tar archive and your extension install script as *ext*. If you want to restore these on another machine, you'd just copy the code-backup directory's contents over and reverse the process shown above (that is, untar the config files into VS Code config directory and run the *ext* script)

Wrapping up

This was just a basic example of backing up your files. If you don't fancy writing backup scripts, you can check out my own full and a little enhanced implementation of this backup workflow over here. There is also a VS Code extension for synchronizing your settings called Settings Sync that synchronizes everything for you into a GitHub Gist.

Radar PROMOTED

The #1 Geolocation Platform

Geofencing, maps, and geolocation compliance, all in a modern, cost-effective platform.

Thousands of companies trust Radar's APIs and SDKs to power location-based experiences across hundreds of millions of devices globally.

Learn more

Top comments (3)

Ky • Mar 17 '23 • Edited

FYI: cfg folder="/home/$USER/.config

/Code/User/" does not exist for me

Excellent article and tutorial. In an effort to be helpful, I added the following "Issue" to your GitHub page regarding your "vscode-backup.sh" file that you shared:

I am not sure if VS Code is supposed to use the path you provided (cfg_folder="/home/$USER/.config/Code/User/") for storing User settings, but mine is not (that path does not exist).

I instead found my "settings.json" here: "/Users/$USER/Library/Application Support/Code"

I then edited the "cfg_folder=..." line to match my above location.

My machine specs, in case that makes a difference or explains something:

```
Type: MacBook Pro 4,1 (17-inch, Early 2008; (MB166LL/A))
Processor: 2.6 GHz Intel Core 2 Duo
Memory: 4 GB 667 MHz DDR2 SDRAM
OS: OS X El Capitan (v. 10.11.6 (15G22010))
```

ALSO, if you care (and are a bit anal like I am), your second line in the intro text (line 5) has a spelling error:

Script prints all intalled VSC extensions --> should be "installed"

Thanks for any comments or feedback you wish to share.

Artur Haurylkevich • Sep 2 '21

```
code --list-extensions | xargs -L 1 echo code --install-extension
```

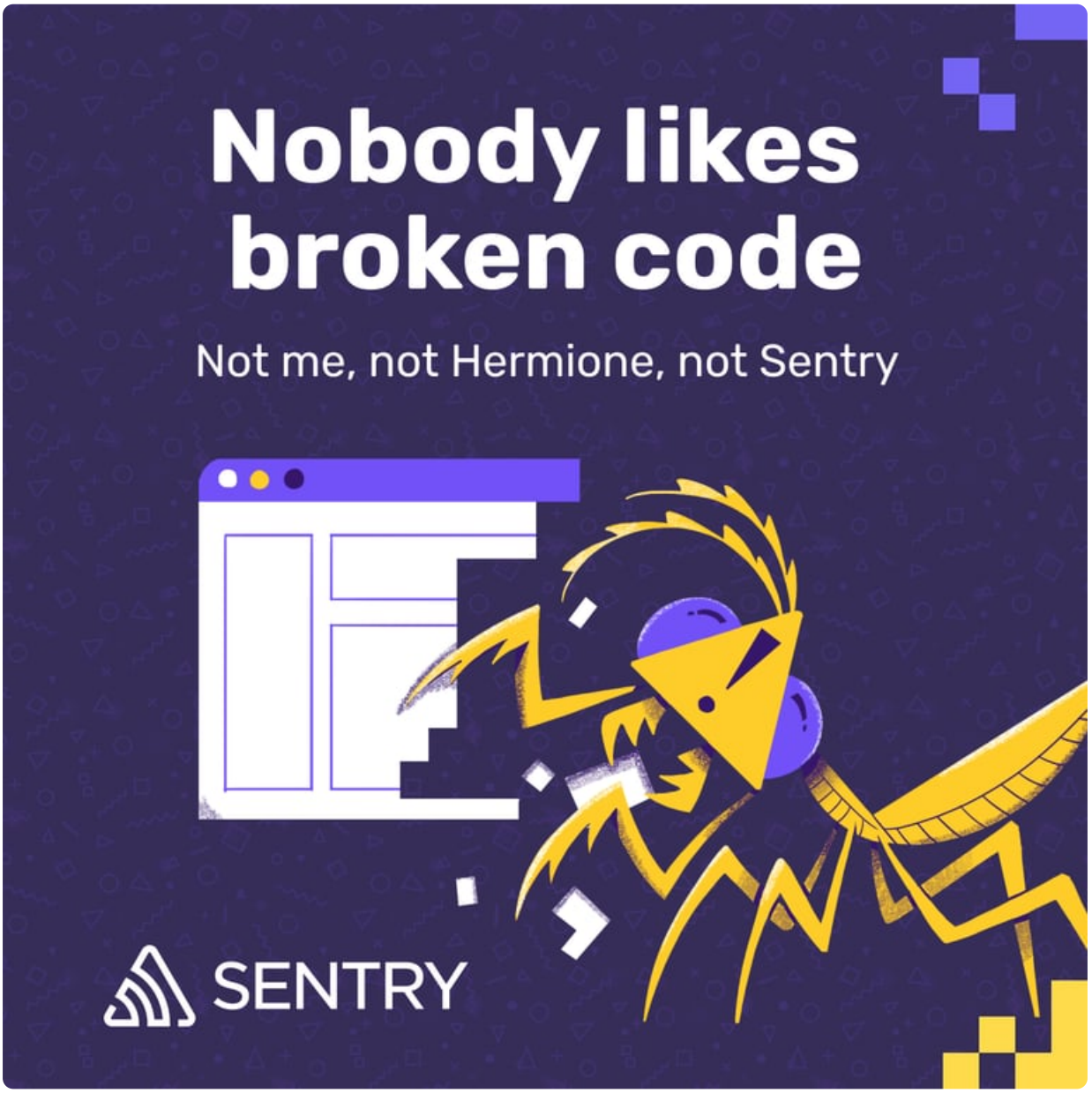
It's a good one. But compressing the json file seems like overhead to me :)

Souris • Jun 28 '21

Thank you, now hopping OS is a bit more quicker :)

Code of Conduct • Report abuse

Sentry PROMOTED



See why 4M developers consider Sentry, “not bad.”

Fixing code doesn’t have to be the worst part of your day. Learn how Sentry can help.

Learn more

Dominik Zarsky

LOCATION
Czech Republic

WORK
DevOps Consultant

JOINED
Feb 22, 2021

Trending on DEV Community

Notpad: Reinventing the Ordinary Notepad Experience with Svelte, Shadcn, and Tauri ✨📝🚀

#svelte #javascript #opensource #productivity

What was your win this week?

#weeklyretro #discuss

Meme Monday

#jokes #discuss #watercooler

AWS PROMOTED

Be bold with AWS Partners at AWS re:Invent

Ready for some innovation inspo? Join AWS and AWS Partners live from Las Vegas, as they cover what's new and noteworthy on the cloud for 2025.

Register now