



Hao's learning log

Coding, Software, Web Development, UX, Productivity, Career tips, Personal Development

- Home
- Search
- Book Recommendations
- About
- RSS



Photo by Drew Beamer on Unsplash

How to use different git emails for personal and work repositories on the same machine

6th October 2020

4 min read

I recently started working on some side hustle projects. To keep things organised, I created a new git account with a different email. When I tried to push to the git repository within the new git account, I got the following error.

```
Enumerating objects: 26, done.
Counting objects: 100% (26/26), done.
Delta compression using up to 12 threads
Compressing objects: 100% (16/16), done.
Writing objects: 100% (21/21), 69.90 KiB | 4.37 MiB/s, done.
Total 21 (delta 2), reused 0 (delta 0)
remote: GitLab: Author 'personal@mail.com' is not a member of team
To gitlab.work.co.uk:project/repo.git
! [remote rejected] feature/a -> feature/a (pre-receive hook declined)
error: failed to push some refs to 'git@gitlab.work.co.uk:project/repo.git'
```

(I removed some sensitive data.)

I didn't want to change my default git email, because I still want to use it for other git repositories. After some research, I've found two ways to solve my problem.

Method 1: use different git email based on repository path

This method involves adding some conditions to global git config. Then within each git repo, it will automatically figure out which email to use based on the path.

By default, the global git config looked something like this for me.



Recent Articles

- GraphQL at scale: schema stitching v.s. schema federation
- A web dev engineer's review: Learning Swift using ChatGPT
- When is GraphQL the right choice for your Project or Business?
- Fixing esbuild related CPU architecture error on Apple silicon Macs
- What I learned about JS Date from writing a simple date library

Tags

- dev-tips
- graphql
- thoughts
- swift
- svelte
- javascript
- front-end
- testing
- career
- miscellaneous
- coding
- typescript
- dev-ops
- back-end
- react
- technology
- productivity
- roam
- gatsby
- coaching
- vue-js
- android
- state-management
- angular-1.x

```
# ~/.gitconfig

[user]
  email = personal@email.com
  name = Hao Dong
```

1. Rename ".gitconfig" to ".gitconfig-personal"

Rename the global config to `~/.gitconfig-personal` using the following command. It will now be our new config for any personal projects.

```
mv .gitconfig .gitconfig-personal
```

Don't worry, we will add the global config back later.

2. Create ".gitconfig-work"

Create the new config file for work.

```
touch ~/.gitconfig-work
```

Then add the following into the config file.

```
# ~/.gitconfig-work

[user]
  email = work@email.com
  name = Hao Dong
```

3. Create a new ".gitconfig"

Add a new global `.gitconfig` file, this will point to the other two custom config files.

```
touch ~/.gitconfig
```

Then add the following into the global config file.

```
# ~/.gitconfig

[includeIf "gitdir:~/work/"]
  path = .gitconfig-work
[includeIf "gitdir:~/personal/"]
  path = .gitconfig-personal
```

Source: https://git-scm.com/docs/git-config#_includes

Method 2: setting git email per repository

Personally, I would suggest going with Method 1, no matter your situation. But if you are confident it is a one-off case, then this method might be appropriate.

Navigate to the work repository, then at the root folder run the following command to change the email.

```
$ git config --local user.email name@work.com
```

Note: this command only affects the current repository. Any other repositories will still use the default email specified in `~/.gitconfig`.

Question: I did all that, but the commit still says the wrong email

If you already committed the code change like I have, then you will have to go through each commit made prior to the email change and then update the email in the commit using the following command.

```
git commit --amend --author="Full Name <email@address.com>" --no-edit
```

■ productivity ■ coding



Buy me a coffee

8 Comments

1

Login

▼

G

Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS



Name



Share

BestNewestOldest



bentinata

9 months ago

Hi.

Came across this article few hours ago when I have some problem with it. Well, didn't find any answer in the internet, but I've found the solution. Decided to write it down here:

<https://git.sr.ht/~bentinata>

Just want to share in case someone has the same problem, since I think the new ``hasconfig:remote.*.url`` is more flexible than ``gitdir:``.

00

Reply

•

Share

›

P

Pierre

a year ago

Hello! I found your blog post a while back and finally decided to give it a go.

The problem I'm facing is that if I'm not located in the directory where `.git` is, the data from `.gitconfig-personal` and `.gitconfig-work` are not used at all...

I'm using git v2.34.1 on Ubuntu 22.04.

After checking the documentation, I tried replacing

`[includeIf "gitdir:~/work/"]` with `[includeIf "gitdir:~/work/**"]`, but got the same result.

If I am at the root of a git repo under my `"~/work"` directory, I get this:

```
$ git config --list
(...)
includeif.gitdir:~/dev/work/**/*.path=.gitconfig-work
user.name=My Name
user.email=my.email@work.com
includeif.gitdir:~/dev/perso/**/*.path=.gitconfig-perso
(...)
```

-> All good.

but if I go into any subdirectory, I get this:

```
$ git config --list
(...)
includeif.gitdir:~/dev/work/**/*.path=.gitconfig-work
includeif.gitdir:~/dev/perso/**/*.path=.gitconfig-perso
(...)
```

-> Not good!

and so, if I try to commit from there, I get an error from git.

Are you facing the same issue? How can I fix this?

00

Reply

•

Share

›

P

Pierre

→ Pierre

a year ago

I found the root cause, and how to work around it.

In my case, `~/dev/` is not a directory, but a symlink that points to another mount

(/mnt/documents/dev). In order to make git work in any subdirs, I replaced

```
[includeIf "gitdir:~/work/"]
```

with

```
[includeIf "gitdir:/mnt/documents/dev/work/"]
```

and did the same for "perso". It now works as expected in any directory of a git repo!

00

Reply • Share ›



Hao Mod

Pierre

a year ago

Sorry for the late reply, glad to see you've got it working! It's probably good that you figured it out yourself, don't think I would have thought of checking symlink 😊

00

Reply • Share ›



Jasurbek Yuldoshev

a year ago

This feature is already available in github, see notifications->custom-routing inside profile settings. No need to do some manual tweaks any more.
But anyway, thank you for the article, I didn't see any other one that is similar with yours!

00

Reply • Share ›



Hao Mod

Jasurbek Yuldoshev

a year ago

Hi Jasurbek, thanks for the reply!

I think what I was trying to do here is different to what you had in mind. I wasn't too concerned with which email the notifications went to. Instead, the purpose of this setup is to have different git configs for work and personal use case. For example, at my current work place I need to sign my commits, there is no way around it. However, I don't want to sign my commits for my own personal repos. Using this setup, I'm able to have my cake and eat it too (signed commits for work repos, and no commit signing for personal ones) 😊

But if all you need is to get Github notification sent to different email addresses, then I 100% agree this is not the best/right method.