Git - How to fix "corrupted" interactive rebase?

Asked 9 years, 1 month ago   Active 3 months ago   Viewed 62k times

▲

118

▼

★

16

I managed to create a little mess in my local git repository. I was trying to fix a broken commit by using the following instructions. Before running the "git commit --amend" (and after the git rebase --interactive) I decided that my changes were incorrect and so I executed "git reset HEAD --hard". Not a good idea, I tell you.

Now the interactive rebase seems to be "stuck". Git shows the current branch as (|REBASE-m). Every command (cd .., ls, git rebase...) inside my repository gives the following error:

> cat: .git/rebase-merge/head-name: No such file or directory

Here's how git rebase --abort looks like:

```
$ git rebase --abort
cat: c:/_work/project/src/git/.git/rebase-merge/quiet: No such file or directory
cat: c:/_work/project/src/git/.git/rebase-merge/head-name: No such file or directory
cat: c:/_work/project/src/git/.git/rebase-merge/orig-head: No such file or directory
HEAD is now at 4c737fb Revert "Modified file names"
rm: cannot remove `c:/_work/project/src/git/.git/rebase-merge/done': Permission denied
rm: cannot remove directory `c:/_work/project/src/git/.git/rebase-merge': Directory
not empty
cat: .git/rebase-merge/head-name: No such file or directory
```

Here's the result of git rebase --continue:

```
$ git rebase --continue
cat: c:/_work/project/src/git/.git/rebase-merge/prev_head: No such file or directory
cat: c:/_work/project/src/git/.git/rebase-merge/end: No such file or directory
cat: c:/_work/project/src/git/.git/rebase-merge/msgnum: No such file or directory
cat: c:/_work/project/src/git/.git/rebase-merge/onto: No such file or directory
cat: c:/_work/project/src/git/.git/rebase-merge/quiet: No such file or directory
prev_head must be defined
cat: .git/rebase-merge/head-name: No such file or directory
```

Any ideas? I would like to reset the situation back to the state it was before I started my well-thought rebase operation.

Here's how git log --oneline shows the situation:

```
4c737fb Revert "Modified file names"
247ac02 Modified file names
33141e6 Message modifications
10a4a04 Modified db script
```

And this is fine.

I'm using msysgit v1.7.0.2.

git   msysgit   git-rebase

edited May 23 '17 at 12:10   asked Sep 10 '10 at 13:37

Community ♦   Mikael Koskinen
1   1   6,892   3   38   59

17 Answers

It looks like Git tried to remove the `.git/rebase-merge` directory but wasn't able to remove it completely. Have you tried copying that folder away? Also copy away the `.git/rebase-apply` folder if that is present.

111

edited Sep 10 '10 at 19:00   answered Sep 10 '10 at 18:47

✓   Martin Owen
                                     3,673   3   28   29

7   Thanks for pointing that out. I was able to fix my problem just by restarting the computer. I'm not sure what went wrong because somehow every access to the .git\rebase-merge folder caused an "Access denied" -error. – Mikael Koskinen Sep 13 '10 at 5:07

17 Reboot did not work for me but `git rebase --abort` (from stackoverflow.com/a/4757777/146044) *did* work. – Backus Mar 28 '16 at 2:24

2   Just restarting the git shell (windows) worked for me (`git rebase --abort` wasn't working) – mhand Mar 31 '16 at 20:47

2   a word of warning... i had 4 hours worth of changes on my working directory when i noticed the corrupted rebase error. tried the `git rebase --abort` it wiped my unstaged changes... the error did disappear though – George Ananda Eman Sep 26 '16 at 12:22 ✎

49 rebooting or `git rebase --abort` still gave me the errors `git rebase --quit` worked for me Flávio Rodrigues Apr 12 at 17:52

49 rebooting or `git rebase --abort` still gave me the errors. `git rebase --quit` worked for me. – Flavio Rodrigues Apr 12 at 17:52

87

I had a similar problem due to a zombie vim.exe process. Killing it in Task Manager, followed by a `git rebase --abort` fixed it.

answered Jan 21 '11 at 10:37

user584317
879   7   3

---

this was my problem too. I used sysinternals `handle` command and saw that a process (sh.exe) had the file locked. Using `pskill <pid>` fixed it for me. – Paul Oliver Aug 17 '15 at 23:07

72

I got stuck in this. I created the head-name file, and then I ran into another error saying it couldn't find the onto file, so I created that file. Then I got *another* error saying could not read '.git/rebase-apply/onto': No such file or directory.

So I looked at the git [documentation](#) for rebasing and found another command called `git rebase --quit`. This set me back on my branch with no changes, and I could start my rebase over again, good as new.

answered Feb 28 at 20:02

Laura Slocum
756   5   6

---

1   Thank you! This fixed mine! – Jeff Lewis Mar 11 at 4:44

1   Double thank you! – Marian Zburlea Jun 20 at 10:30

13   `git rebase --quit` was it! – Steven Shaw Jun 25 at 1:14

1   `--quit` flag FTW – Kraken Jul 29 at 8:38

1   `--quit` worked. `--abort` didn't because the rebase was cancelled half way through – Kalob Taulien Sep 27 at 21:51

13

Thanks @Laura Slocum for your answer

I messed things up while rebasing and got a detached HEAD with an

```
error: could not read orig-head
```

that prevented me from finishing the rebasing.

The detached HEAD seem to contain precisely my correct rebase desired state, so I ran

```
rebase --quit
```

and after that I checked out a new temp branch to bind it to the detached head.

By comparing it with the branch I wanted to rebase, I can see the new temp branch is exactly in the state I wanted to reach. Thanks

answered Jul 12 at 8:49

Ruth
146　1　7

7

Had same problem in Eclipse. Could not Rebase=>abort from Eclipse.

Executing **git rebase --abort** from Git Bash Worked for me.

answered Apr 4 '14 at 6:46

Praveen Hiremath
3,696　2　12　10

6

On Windows, if you are unwilling or unable to restart the machine see below.

Install Process Explorer: https://technet.microsoft.com/en-us/sysinternals/bb896653.aspx

In Process Explorer, Find > File Handle or DLL ...

Type in the file name mentioned in the error (for my error it was 'git-rebase-todo' but in the question above, 'done').

Process Explorer will highlight the process holding a lock on the file (for me it was 'grep').

Kill the process and you will be able to abort the git action in the standard way.

answered Mar 27 '15 at 10:32

Darren Yeats

🟥 **71** 1 2

▲

4

```
touch .git/rebase-merge/head-name
```

▼

and than use `git rebase`

edited Apr 6 '15 at 15:39   answered Apr 6 '15 at 15:20
Ethaan   Tehila Dabush
10.5k 5 28 42   **41** 3

▲

3

In my case eighter `git rebase --abort` and `git rebase --continue` was throwing:

> error: could not read '.git/rebase-apply/head-name': No such file or directory

▼

I managed to fix this issue by manually removing: `.git\rebase-apply` directory.

answered Feb 20 at 12:25
Dariusz
8,364 5 42 63

▲

1

I have tried all the above steps mentioned but nothing worked for me. Finally, **restarting** the computer worked for this issue :D

answered Jan 3 at 5:34
Mohammed Asif
16 2

▼

▲

I'm using `git version 2.19.2.windows.1 .`

the only thing that worked for me was to remove the `.git/rebase-apply/` directory and do a `git reset --hard` .

1

answered Dec 6 '18 at 19:07

Spongman
5,724 6 29 49

In my case it was because I had opened SmartGit's Log in the respective Git project and Total Commander in the respective project directory. When I closed both I was able to rebase without any problem.

1

The more I think about it, the more I suspect Total Commander, i.e. Windows having a lock on opened directory the git rebase was trying to something with.

Friendly advice: When you try to fix something, always do one change at a time. ;)

edited Mar 9 '16 at 8:58   answered Mar 9 '16 at 8:51

meridius
719 10 25

With SublimeText 3 on Windows, the problem is fixed by just closing the Sublime windows used for interactive commit edition.

1

answered Mar 28 at 10:13

LeBorgne
69 8

I had the same problem. I used process explorer as suggested in other post (i am not able to find that post) and figured out which process has a lock on the file and kill it. then execute the --continue or --abort as per needs

0

answered Jul 9 '14 at 0:38

user3739537
28 2 7

Once you have satisfactorily completed rebasing X number of commits , the last command must be `git rebase --continue` . That

completes the process and exits out of the rebase mode .

0

edited Nov 21 '13 at 23:10   answered Oct 24 '13 at 16:25
Karel Bílek   diptia
15k   24   70   127   **1,576**   16   19

In my case after testing all this options and still having issues i tried `sudo git rebase --abort` and it did the whole thing

0

answered May 3 '16 at 13:34
Dani.Rangelov
326   2   5

> Be careful if you're doing this in the middle of an actual rebase. Just lost my changes :'(– Freeman Latif Sep 11 '17 at 10:05

> It was considered "corrupted", then you start over with the rebase – Dani.Rangelov Sep 12 '17 at 7:19

---

tried everything else but a reboot, what worked for me is `rm -fr .git/REBASE_HEAD`

0

answered May 7 at 12:59
Jeremy Vaught
11   1

I am using git in eclipse and I was having the same problem.

0

Eventually I found that the "Rebase ..." menu entry was temporarily transformed into a sub-menu.

Team-> Rebase -> Abort

It worked for me.

answered Oct 14 '14 at 12:57
Shiva Agrawal
143   2   9