

# Issues

## fsck: dangling commit

List details about dangling commits ([Git - Finding Lost/Dangling Commits](#)):

```
git fsck --lost-found | grep "^dangling commit" | sed "s/^dangling commit  
//g" | xargs git show -s --oneline
```

Cleanup:

```
git reflog expire --expire=now --all  
git gc --prune=now --aggressive  
git repack -a -d
```

## Reduce repo size

It is important to check if git configured with default settings. Especially forcing of creation of multiple smaller packs significantly increases repository size. I.e. changing one 1.2G pack into multiple 256MB packs let, increases size of packs to 4.3GB :(

```
git config --global -l  
git config --system -l
```

Especially take attention for options like:

- deltaCacheSize = 10m
- packSizeLimit = 10m
- windowMemory = 10m

Command

```
git gc
```

and other cleaning commands requires whole repository to be read into memory. For big repos (like kernel.git) it requires to allocate over 8GB of RAM. Compression of objects is also performed in RAM, then packs are written onto disc.

<https://stackoverflow.com/questions/42175296/git-gc-uses-a-lot-of-memory-even-i-limited-it>

To prevent touching existing packs and exhaust system memory paramter auto should be used.

```
git gc --auto
```

Another possibility is to mark some packs as keep by creating .keep files.

Remove unreachable objects:

```
git count-objects -v
git gc --auto # this is enough to remove loose objects and repack repo
git count-objects -v

# more instruction
git repack -Ad # Remove dangling objects from packfiles
git prune # Remove dangling loose objects
git gc --aggressive --prune=now --force || rm -f .git/objects/*/tmp_* && rm
-f .git/objects/*/tmp-*

git count-objects -v
```

## error: pack-objects died of signal 9

git was killed by host monitor process or oom due to memory usage. Check syslog for OOM entries.

Solution:

```
[core]
    packedGitLimit = 32m
    packedGitWindowSize = 32m
    deltaCacheSize = 10m
```

Disable compression:

```
git config core.compression 0
git config core.loosecompression 0
git config pack.window 0
```

### gitlab

git command running under user 'git' with gitlab installed looks for configuration in following places:

- current git repo
- /etc/gitconfig
- /var/opt/gitlab/.config/git/config
- **/var/opt/gitlab/.gitconfig**

This is a user config file, and file is generated from  
/opt/gitlab/embedded/cookbooks/gitlab/templates/default/gitconfig.erb

But if git binary from gitlab is used

```
sudo -u git -H "/opt/gitlab/embedded/bin/git" config -l
```

```
/opt/gitlab/embedded/bin/git --git-dir=/home/git-  
data/repositories/funny/linux.git config -l  
/opt/gitlab/embedded/bin/git --git-dir=/home/git-  
data/repositories/funny/linux.git bundle create /tmp/linux.bundle --all
```

then, file /opt/gitlab/embedded/etc/gitconfig is used. To modify this file:

gitlab.rb

```
omnibus_gitconfig['system'] = {.  
  "core"      => ["packedGitLimie = 32m", "packedGitWindowSize = 32m",  
"deltaCacheSize = 32m" ],  
  "receive"   => ["fsckObjects = true"],  
  "pack"      => ["deltaCacheSize = 32m", "packSizeLimit = 32m",  
"windowMemory = 32m", "threads = 2" ]  
}
```

Test config:

```
gitlab-ctl show-config
```

Regenerate config:

```
gitlab-ctl reconfigure
```

```
sudo -u git bash  
cd /var/opt/gitlab/git-data/repositories/{namespace}/{repository}.git  
git gc --aggressive
```

From:  
<https://niziak.spox.org/wiki/> - niziak.spox.org

Permanent link:  
<https://niziak.spox.org/wiki/git:issues>

Last update: **2023/07/17 16:02**

