

Git

Test Scenarios

Summary:

- When cloning from remote to local, www-data group ownership is not preserved.
- When pushing local changes to remote, need to checkout a different branch on remote first, and permissions are not preserved.

Initial Cloning

- Create two files in a remote folder to simulate the wiki install on the server.
- Change their permissions and ownership to match what's on the server:

```
~/gittest/remote$ ls -l
total 24
-rwxrwx--- 1 ptruchon www-data 26 Dec 26 10:20 file1.txt
-rwxrwx--- 1 ptruchon www-data 26 Dec 26 10:20 file2.txt
```

- Initialize git:

```
git init
git add .
git commit -m 'initial commit on remote'
git log
commit 0e5ccdedd1c9188996aadad3a8e2be5a319ab789 (HEAD -> master)
Author: Patrick Truchon <patoo@rbox.me>
Date: Sun Dec 26 10:21:24 2021 -0800

    initial commit on remote
```

- Create a local folder and clone from remote:

```
git clone ~/gittest/remote/ ~/gittest/local/
```

- The log on the local copy looks good:

```
~/gittest/local$ git log
commit 0e5ccdedd1c9188996aadad3a8e2be5a319ab789 (HEAD -> master, origin/master,
origin/HEAD)
Author: Patrick Truchon <patoo@rbox.me>
Date: Sun Dec 26 10:21:24 2021 -0800

    initial commit on remote
```

- But the `www-data` group ownership and the 770 permissions weren't preserved:

```
~/gittest/local$ ls -l
total 24
-rwxrwxr-x 1 ptruchon ptruchon 26 Dec 26 10:24 file1.txt
-rwxrwxr-x 1 ptruchon ptruchon 26 Dec 26 10:24 file2.txt
```

- Change the group back to `www-data` and permissions back to 770:

```
sudo chgrp www-data *; chmod 770 *
ls -l
total 24
-rwxrwx--- 1 ptruchon www-data 26 Dec 26 10:24 file1.txt
-rwxrwx--- 1 ptruchon www-data 26 Dec 26 10:24 file2.txt
```

Making Local Changes

- Edit `file1.txt` from the local directory.

```
~/gittest/local$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   file1.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

- Commit changes in git:

```
git add .
git commit -m 'changed file1.txt from local'
git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

git log
commit b7d91058fed9fe64c96525a3d0bef56c682ade68 (HEAD -> master)
Author: Patrick Truchon <patoo@rbox.me>
Date:   Sun Dec 26 10:45:41 2021 -0800

    changed file1.txt from local

commit 0e5ccdedd1c9188996aadad3a8e2be5a319ab789 (origin/master, origin/HEAD)
Author: Patrick Truchon <patoo@rbox.me>
```

Date: Sun Dec 26 10:21:24 2021 -0800

initial commit on remote

Push Changes to Remote

- Attempting to push changes to remote while master is still checked out on remote doesn't work:

```
~/gittest/local$ git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 329 bytes | 329.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote: error: refusing to update checked out branch: refs/heads/master
remote: error: By default, updating the current branch in a non-bare repository
remote: is denied, because it will make the index and work tree inconsistent
remote: with what you pushed, and will require 'git reset --hard' to match
remote: the work tree to HEAD.
remote:
remote: You can set the 'receive.denyCurrentBranch' configuration variable
remote: to 'ignore' or 'warn' in the remote repository to allow pushing into
remote: its current branch; however, this is not recommended unless you
remote: arranged to update its work tree to match what you pushed in some
remote: other way.
remote:
remote: To squelch this message and still keep the default behaviour, set
remote: 'receive.denyCurrentBranch' configuration variable to 'refuse'.
To /home/ptruchon/gittest/remote/
! [remote rejected] master -> master (branch is currently checked out)
error: failed to push some refs to '/home/ptruchon/gittest/remote/'
```

- Workaround is to checkout a temporary branch on remote:

```
~/gittest/remote$ git checkout -b tmp
Switched to a new branch 'tmp'
```

- Then push the local changes to remote:

```
~/gittest/local$ git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 329 bytes | 329.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To /home/ptruchon/gittest/remote/
```

```
0e5ccde..b7d9105 master -> master
```

- Check status and log from local:

```
~/gittest/local$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean

~/gittest/local$ git log
commit b7d91058fed9fe64c96525a3d0bef56c682ade68 (HEAD -> master, origin/master,
origin/HEAD)
Author: Patrick Truchon <patoo@rbox.me>
Date:   Sun Dec 26 10:45:41 2021 -0800

    changed file1.txt from local

commit 0e5ccdedd1c9188996aadad3a8e2be5a319ab789
Author: Patrick Truchon <patoo@rbox.me>
Date:   Sun Dec 26 10:21:24 2021 -0800

    initial commit on remote
```

- On remote, checkout master and delete the temporary branch

```
~/gittest/remote$ git checkout master; git branch -d tmp
Switched to branch 'master'
Deleted branch tmp (was 0e5ccde).
```

- Status and log both look good:

```
git status
On branch master
nothing to commit, working tree clean

git log
commit b7d91058fed9fe64c96525a3d0bef56c682ade68 (HEAD -> master)
Author: Patrick Truchon <patoo@rbox.me>
Date:   Sun Dec 26 10:45:41 2021 -0800

    changed file1.txt from local

commit 0e5ccdedd1c9188996aadad3a8e2be5a319ab789
Author: Patrick Truchon <patoo@rbox.me>
Date:   Sun Dec 26 10:21:24 2021 -0800

    initial commit on remote
```

- But group ownership of the modified file wasn't preserved

```
~/gittest/remote$ ls -l
total 24
-rwxrwxr-x 1 ptruchon ptruchon 50 Dec 26 12:01 file1.txt
-rwxrwx--- 1 ptruchon www-data 26 Dec 26 10:20 file2.txt
```