

Gitosis

Christian Külker

2024-02-28

Contents

1	Introduction	1
2	Manual Installation [0.1.0]	2
3	Creating a New Repository [0.1.0]	3
3.1	Adding Access For A New Client	3
3.2	Configuring and Using the Git Repository	3
4	Global Access [0.1.0]	5
5	Access to GitWeb [0.1.0]	5
6	Rename a gitosis Repository [0.1.0]	6
7	Remove a Repository [0.1.0]	7
8	Use A Git Hook [0.1.0]	7
9	Other Links	7
10	History	7
11	Disclaimer of Warranty	8
12	Limitation of Liability	8

1 Introduction

This old document from 2011 describes **gitosis**, software to manage and deploy git repositories. It provides an easy and stable model to access git repositories for teams using a shared server. As of Debian 7.x (Wheezy), gitosis has been replaced by [Gitolite](#). Therefore,

this document will most likely not receive substantial updates and is for historical purpose only.

2 Manual Installation [0.1.0]

It is relatively easy to manually install **gitosis**. This installation will install it to `/home/srv` (user data) to `/srv` (short path). It uses two computers: (a) server and (b) client.

```
# On the server as root:
rmdir /srv
mkdir /home/srv
ln -s /home/srv /srv

aptitude install gitosis
```

Make sure this user exists

```
id gitosis
uid=104(gitosis) gid=107(gitosis) groups=107(gitosis)
```

The most important locations are:

```
/home/srv/gitosis
/home/srv/gitosis/git
/usr/bin/gitosis-init
/usr/bin/gitosis-run-hook
/usr/bin/gitosis-serve
```

For the initial setup, we need one SSH key. You can use the key of your current user on the **client**.

```
# on the client
cd
scp .ssh/id_rsa.pub rep:/tmp
```

You need to be `root` first, and then change to the user `gitosis`

```
# As root on the server:
mkdir /srv/gitosis # sometimes
su - gitosis
gitosis-init < /tmp/id_rsa.pub
Initialized empty Git repository in
↳ /home/srv/gitosis/repositories/gitosis-admin.git/
```

```
Reinitialized existing Git repository in
↳ /home/srv/gitosis/repositories/gitosis-admin.git/
```

On the client

```
git clone gitosis@rep.example.com:gitosis-admin.git
```

Then, you edit the gitosis-admin repository on your client.

3 Creating a New Repository [0.1.0]

This will create the repository `vwtg` on `m.example.com`.

```
export REP=vwtg
mkdir $REP
cd $REP
git init
echo "gitosis@m.example.com:$REP.git" >| REP
git add REP
git commit -m init REP
git remote add origin gitosis@m.example.com:$REP.git
git config branch.master.remote origin
git config branch.master.merge refs/heads/master
git push origin master:refs/heads/master
git pull
```

3.1 Adding Access For A New Client

The access to **gitosis** is based on public SSH keys.

1. Add pub key to keydir
2. Name it after the login and machine name, usually domain name, for example `user@machine2.pub`
3. Add that string `user@machine2` to the 'members' section like so:

```
members = user@machine1 user@machine2
```

3.2 Configuring and Using the Git Repository

This example creates the repository `hpc-benchmark-suite` and assign it to the group `hpc-sw-dev` inside the configuration in the first step and then in the second step it ac-

tually created the repository. In a third step it adds configuration to the repository.

1. Add configuration to `gitosis.conf`

```
1 [repro hpc-benchmark-suite]
2 description = Aurora Benchmark Suite
3 owner = user_id
4
5 [group hpc-sw-dev]
6 writable = hpc-benchmark-suite
7 members = user_id
```

2. Create locally a git repository

```
mkdir hpc-benchmark-suite
cd hpc-benchmark-suite
git init
touch README
git add README
git commit -m init README
```

3. Add a remote origin

```
git remote add origin gitosis@rep.example.com:hpc-benchmark-suite.git
#git remote add origin gitosis@m.example.com:nagios.git
```

Or

```
git remote add origin gitosis@m:hpc-benchmark-suite.git
```

If you already have a remote origin, edit `.git/config`

Then push something

```
git push origin master:refs/heads/master
```

To pull we need to update the configuration

```
git config branch.master.remote origin
git config branch.master.merge refs/heads/master
```

4 Global Access [0.1.0]

In addition to the usual restricted access, it is possible to give access to a given repository to all.

```
# As root or gitosis user (depending on access rights)
cd /srv/gitosis/repositories/hpc-benchmark-suite.git
touch git-daemon-export-ok
/usr/lib/git-core/git-daemon --base-path=/srv/gitosis/repositories/
```

Global access is then possible for some directories as:

```
git clone rep.example.com:hpc-benchmark-suite.git
```

However it is still possible to access it as a specific user.

```
git clone gitosis@rep.example.com:hpc-benchmark-suite.git
```

5 Access to GitWeb [0.1.0]

Installation

```
aptitude install gitweb
```

Edit `/etc/gitweb.conf`

```
sed -ie 's%/var/cache/git%/srv/gitosis/repositories%' /etc/gitweb.conf
echo '$export_ok = "git-daemon-export-ok";' >> /etc/gitweb.conf
```

```
1 [gitosis]
2 gitweb = yes
```

Then go to the directory of the project you want to be visible in gitWeb

```
cd /srv/gitosis/repositories/hpc-benchmark-suite.git
touch git-daemon-export-ok
```

Finally start the GitWeb daemon as **gitosis** user

```
/usr/lib/git-core/git-daemon --base-path=/srv/gitosis/repositories/
```

Go to the URL and verify: <http://rep/gitWeb/>

There is a known problem, that the started daemon vanishes from the process list and/or the file `/srv/gitosis/repositories/<REPO>.git/git-daemon-export-ok` gets removed (and the daemon stops).

To mitigate this a `daemon = yes` and `gitWeb = yes` line has to added to `gitosis.conf` for that repository.

```
1 daemon = yes
2 gitweb = yes
```

6 Rename a gitosis Repository [0.1.0]

The rename a `gitosis` repository the order of actions is important.

Before

```
1 [group main]
2 writable = nagios
```

After

```
1 [group main]
2 writable = nagios3
```

Push changes

```
git push origin master
```

Connect to the `gitosis` server and rename the correct folder

```
cd /home/srv/gitosis/repositories
mv nagios nagios3
```

Change the remote reference in all repository clones

```
cd /srv
mv nagios nagios3
cd nagios3
git remote rm origin
git remote add origin gitosis@m:nagios3.git
```

```
git config branch.master.remote origin
git config branch.master.merge refs/heads/master
```

7 Remove a Repository [0.1.0]

The order of actions to remove an existing repository is important. This example uses the git repository `sai` and changes need to be done in `gitosis.conf`. Remove all `sai` entries, as `gitosis admin` user, and then as root.

As the last step remove the repository.

```
cd /srv/gitosis/repositories
rm -rf sai.git
```

8 Use A Git Hook [0.1.0]

To execute an action for every change on the server, add a git hook inside the repository.

```
cd /srv/gitosis/repositories/christian.git/hooks
vim post-receive
```

```
#!/bin/bash
unset $(git rev-parse --local-env-vars)
DATE=`date +%FT%T`
echo $DATE >/tmp/git-hook.log
```

9 Other Links

- <https://wiki.archlinux.org/index.php/Gitosis>

10 History

Version	Date	Notes
0.1.1	2024-02-28	Release to GitHub, update Markdown formatting
0.1.0	2011-07-14	Initial release

11 Disclaimer of Warranty

THERE IS NO WARRANTY FOR THIS INFORMATION, DOCUMENTS AND PROGRAMS, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE INFORMATION, DOCUMENT OR THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE INFORMATION, DOCUMENTS AND PROGRAMS IS WITH YOU. SHOULD THE INFORMATION, DOCUMENTS OR PROGRAMS PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12 Limitation of Liability

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE INFORMATION, DOCUMENTS OR PROGRAMS AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE INFORMATION, DOCUMENTS OR PROGRAMS (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE INFORMATION, DOCUMENTS OR PROGRAMS TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.