git repo server

seguridad

- **git-shell**
  - `usermod -s /usr/bin/git-shell <username>`
- **/etc/ssh/sshd_config**:

```
Match User git
    ForceCommand /usr/bin/git
    AllowTCPForwarding no
    X11Forwarding no
```

- **gitlab-ce**:

  ~git/.ssh/authorized_keys

```
command="/opt/gitlab/embedded/service/gitlab-shell/bin/gitlab-shell key-1",no-port-forwarding,no-X11-forwarding,no-agent-forwarding,no-pty ssh-rsa AA...restofkeysnip!
```

  - no **~/.bashrc**, **~/.profile**
- https://unix.stackexchange.com/questions/489771/block-git-user-from-login-but-allow-the-user-to-still-function-as-a-git-repo
- https://git-scm.com/book/en/v2/Git-on-the-Server-The-Protocols#The-HTTP/S-Protocol

repositorio

https://www.linux.com/training-tutorials/how-run-your-own-git-server/

crear repositorio bare

```
mkdir repo.git
git --git-dir=repo.git init %%--%%bare
```

crear repositorio para compartir

```
git init --bare --shared=group repo_path
chgrp -R foo repo_path
```

convertir repositorio a su versión bare

```
mv repo/.git repo.git
git --git-dir=repo.git config core.bare true
rm -rf repo
```

```
git clone --bare repo_path bare_repo_path
```

```
git clone --mirror repo_path bare_repo_path
```

Set up a mirror of the source repository. This implies --bare. Compared to --bare, --mirror not only maps local branches of the source to local branches of the target, it maps all refs (including remote-tracking branches, notes etc.) and sets up a refspec configuration such that all these refs are overwritten by a git remote update in the target repository. https://git-scm.com/docs/git-clone