Yes, you can do

git push <https://username:password@myrepository.biz/file.git> --all

in this case <https://username:password@myrepository.biz/file.git> replace the origin in git push origin --all

To see more options for git push, try git help push

From <[*https://stackoverflow.com/questions/29776439/username-and-password-in-command-for-git-push*](https://stackoverflow.com/questions/29776439/username-and-password-in-command-for-git-push)>

$ git config credential.helper store  
$ git push <https://github.com/repo.git>

Username for '<https://github.com>': <USERNAME>  
Password for '<https://USERNAME@github.com>': <PASSWORD>

From <[*https://stackoverflow.com/questions/6565357/git-push-requires-username-and-password*](https://stackoverflow.com/questions/6565357/git-push-requires-username-and-password)>

$ git clone <https://github.com/lakshman-a/SHPF.git>

$ git push <https://github.com/lakshman-a/SHPF.git>

fatal: HttpRequestException encountered.

An error occurred while sending the request.

Username for '<https://github.com>': lakshman-a

Counting objects: 3, done.

Delta compression using up to 4 threads.

Compressing objects: 100% (2/2), done.

Writing objects: 100% (3/3), 296 bytes | 0 bytes/s, done.

Total 3 (delta 1), reused 0 (delta 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To <https://github.com/lakshman-a/SHPF.git>

5658af0..89970a4 master -> master

GCP install JAVA

1. Install JAVA
2. <https://www.linode.com/docs/development/java/install-java-on-ubuntu-16-04/>

GCP:

pp-devcos-test-automation

Ubuntu VM Docker:

10.176.17.29

Now we need to Run the Hub using below command

docker run -d --name selenium-hub selenium/hub

*From <*[*https://www.seleniumeasy.com/selenium-tutorials/configure-selenium-grid-using-docker*](https://www.seleniumeasy.com/selenium-tutorials/configure-selenium-grid-using-docker)*>*

If you want to assign the port, run the below command

Not use below 2 lines

//docker run -d -p 4545:4444 --name selenium-hub selenium/hub

//docker run -d -p 4444:4444 --net grid --name selenium-hub

To check if the container has started

*Docker ps*

To check the configuration information, run

docker logs

############## RUN HUB AND NODE ###################

docker run --rm=true -P -p 4444:4444 --name selenium-hub selenium/hub

Console:

<http://10.176.17.29:4444/grid/console>

sudo docker run -d --link selenium-hub:hub selenium/node-firefox

sudo docker run -d --link selenium-hub:hub selenium/node-chrome

layyakannu@instance-1:~$ docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

872f1404b145 selenium/node-chrome "/opt/bin/entry\_poin…" 5 seconds ago Up 4 seconds unruffled\_bohr

9b4120fbd128 selenium/node-firefox "/opt/bin/entry\_poin…" 3 minutes ago Up 3 minutes gallant\_easley

b035b2928d07 selenium/hub "/opt/bin/entry\_poin…" 13 minutes ago Up 12 minutes 0.0.0.0:4444->4444/tcp selenium-hub

Machine generated alternative text:
vwodyrew commented on Nov 5, 201 S 
have the same issue, am using forever wrongly? 
14 
e:eø 
14:ea 
14:14 
S forever 
vø.ls.l 
- -version 
S forever start -c "npm start" /some/dir 
bash 
info: 
Forever processing file: /some/dir/ 
S forever list 
info: 
data : 
data : 
Forever processes running 
uid command sc ript 
forever pid id logfile 
uptime 
9T8K npm start /sone/dir/ 031 
8048 /home/me,' . log 371 
S forever stop 
info: 
Forever stopped process: 
uid command sc ript 
forever pid id logfile 
uptime 
9T8K npm start /some/dir/ 031 
8348 
S forever list 
/ home/me/ . forever/9T8K . log 
info: 
No forever processes running 
S ps aux I grep 
8065 
8066 
8531 
'forever\ node' 
a 4444 660 
a.4 705192 377ee 
a. a 15944 944 pts/27 
sh -c ./modules r 
node . 
- -color—auto forever\lnode 
grep 
ps kill 8065 leaves 8066 running and script is still responding via http. 
ps kill 8066 removes both processes and brings down the http server. 
@ I hope this information is useful. 
Forever lives up to its name! 