**GIT LAB**

**Supported platform.**

* Ubuntu
* Debian
* Centos
* OpenSUSE
* RHEL
* Scientific/oracle linux

NOTE - doesn’t directly support windows, workaround is to user docker container.

Refer - <https://www.tutorialspoint.com/gitlab/gitlab_installation.htm>

**System Requirement.**

* Min 1 core cpu
* 4GB ram.
* Database – (postgresql(highly recommended)/Mysql/mariadb)

**GitLab conditions for file and repo.**

* Maximum 10mb for each file.
* Maximum 10gb per repository

**Installation in RHEL (refer the below URL)**

<https://about.gitlab.com/installation/#centos-6>

**Installation in UBUNTU (refer the below URL)**

<https://about.gitlab.com/installation/#ubuntu>

**NOTE:** All configurations are store inside gitlab.rb file. (/etc/gitlab/gitlab.rb)

**Installation in RHEL 6.9 steps**

First make the normal user as sudo user by editing the file

*/etc/sudoers*

Add the below line in the root section of the file

<user\_name> ALL=(ALL) NOPASSWD:ALL

Below are the application which we install as pre-req.

sudo yum install -y curl policycoreutils-python openssh-server cronie

To open SSH and HTTP firewall rulesets we need to pass the command.

sudo lokkit -s http -s ssh

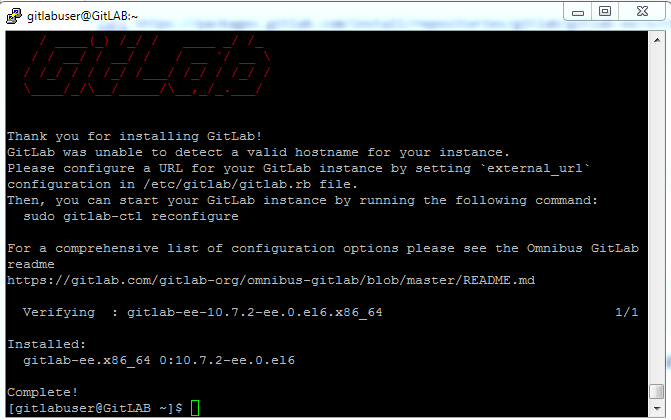
NOTE – read about lokkit

To update the repo with gitlab package, run the command,

curl https://packages.gitlab.com/install/repositories/gitlab/gitlab-ee/script.rpm.sh | sudo bash

sudo EXTERNAL\_URL="http://gitlab.example.com" yum -y install gitlab-ee

The above command will install and set the URL with which we will access gitlab.

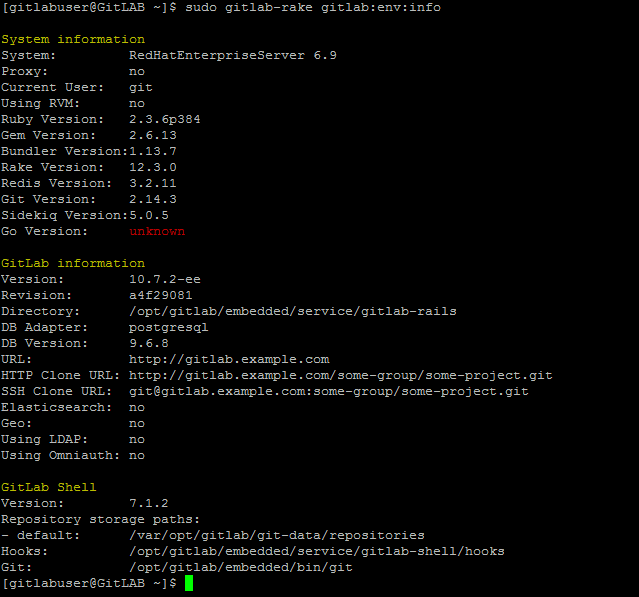


**Post installation**

Gitlab-ctl reconfigure

**To check the version of gitlab with all its dependencies**

Sudo gitlab-rake gitlab:env:info



**How to use https instead of http to secure gitlab**

\*this is pending to be tested not working at the moment\*

Generating self-sign certificate without signing.

Step 1 – Generate the key file.

openssl genrsa -out "/etc/pki/tls/private/gitlab-registry.key" 4096

Step 2 – Generate the cert file.

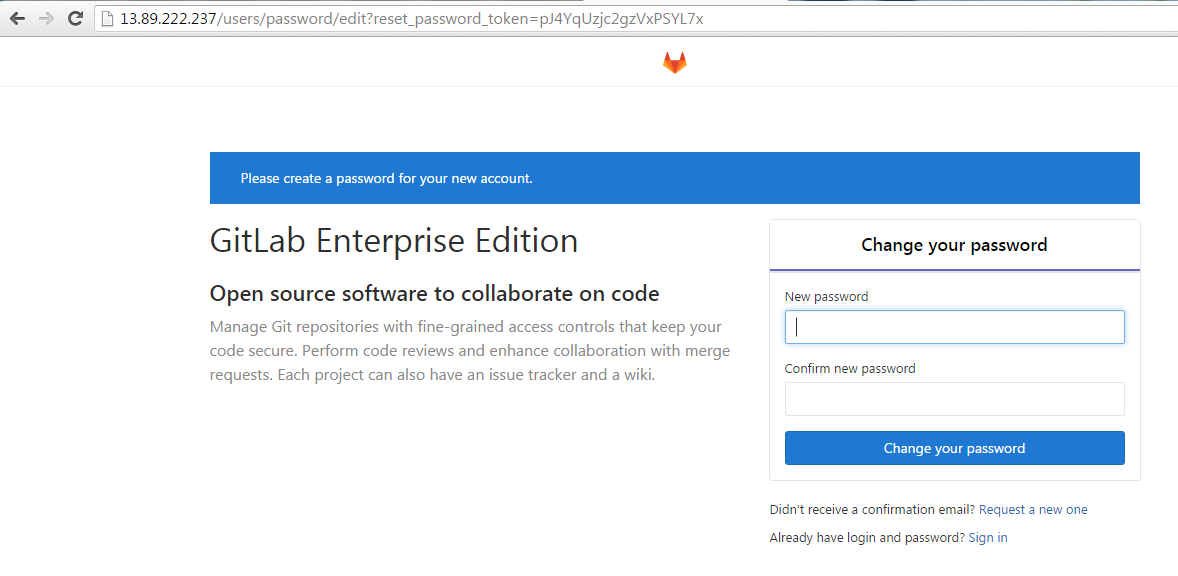
openssl req -x509 -sha512 -nodes -newkey rsa:4096 -days 365 -keyout /etc/pki/tls/private/gitlab.key -out /etc/pki/tls/certs/gitlab.cr

Edit the file /etc/gitlab/gitlab.rb



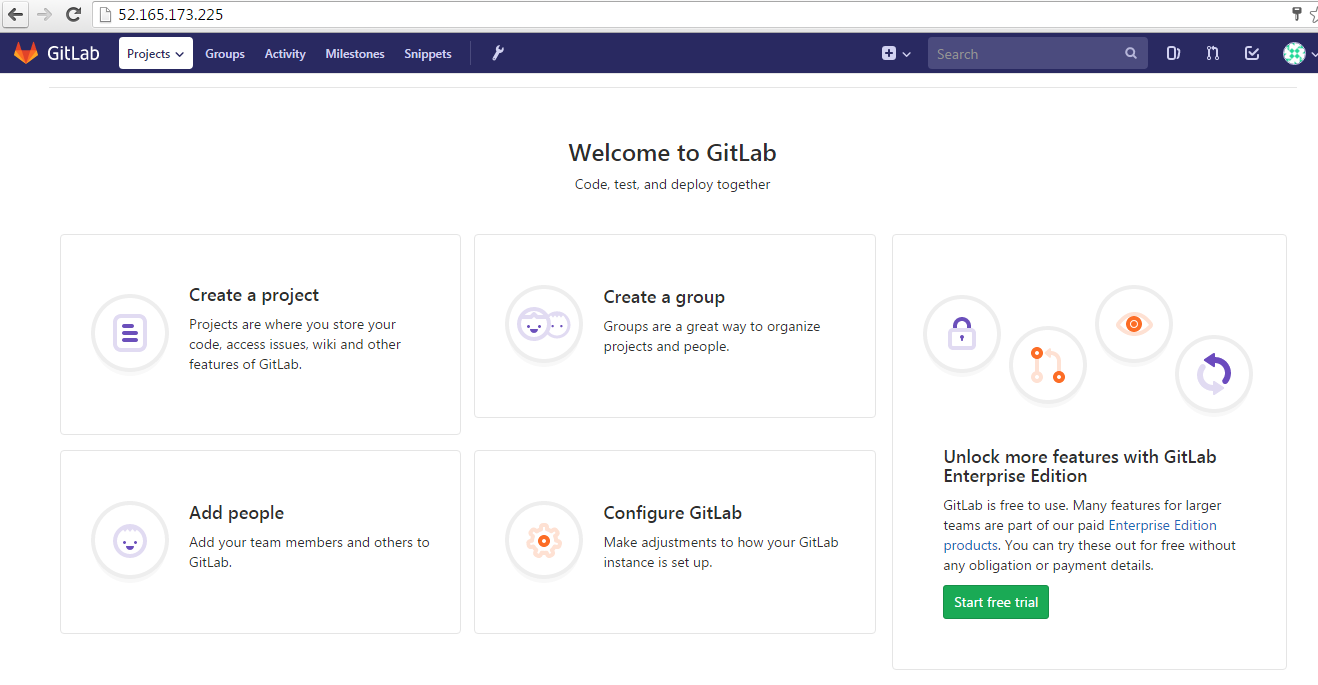
**Access the gitlab page from the browser with the machine public IP**

First time accessing the URL will ask to set the admin password as shown below.



After changing password it will prompt for user name and password.

Username will be **root.**



**Configuring email in gitlab**

<https://docs.gitlab.com/omnibus/settings/smtp.html>

After configuring you can test using the below rub rails command.

gitlab-rails console

Notify.test\_email('eswarraju.konduru@gmail.com', 'test IP', 'hello').deliver\_now

add the below lines to gitlab.rb

gitlab\_rails['smtp\_enable'] = true

gitlab\_rails['smtp\_address'] = "smtp.gmail.com"

gitlab\_rails['smtp\_port'] = 587

gitlab\_rails['smtp\_user\_name'] = "juber.ahmed14@gmail.com"

gitlab\_rails['smtp\_password'] = “<password>”

gitlab\_rails['smtp\_domain'] = "smtp.gmail.com"

gitlab\_rails['smtp\_authentication'] = "login"

gitlab\_rails['smtp\_enable\_starttls\_auto'] = true

gitlab\_rails['smtp\_tls'] = false

gitlab\_rails['gitlab\_email\_from'] = "juber.ahmed14@gmail.com"

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Where does git stores its data.**

Git stores its data in the below location by default.

*/var/opt/gitlab/git-data/repositories/root*

Where “root” is the user with which we logged in to gitlab.

**How to change the storage directory in gitlab.**

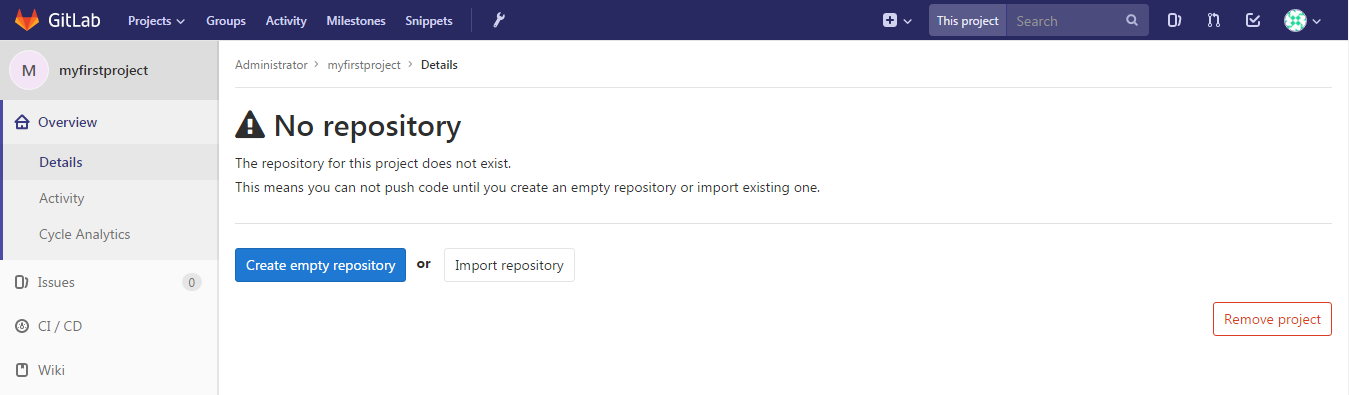
In the gitlab machine look for the drive you want to store the data,   
To figure out the drive with largest space type “df –h “ and select the path if you have any LVM or any drive.

Edit the file /etc/gitlab/gitlab.rb and add the below.

git\_data\_dirs({   
   "default" => {   
     "path" => "/app/opt/git-data"   
    }

}) //give any directory in the path variable

NOTE: the existing repo from the previous dir will get imported into the new location, it will ask to import as shown below when you access it.



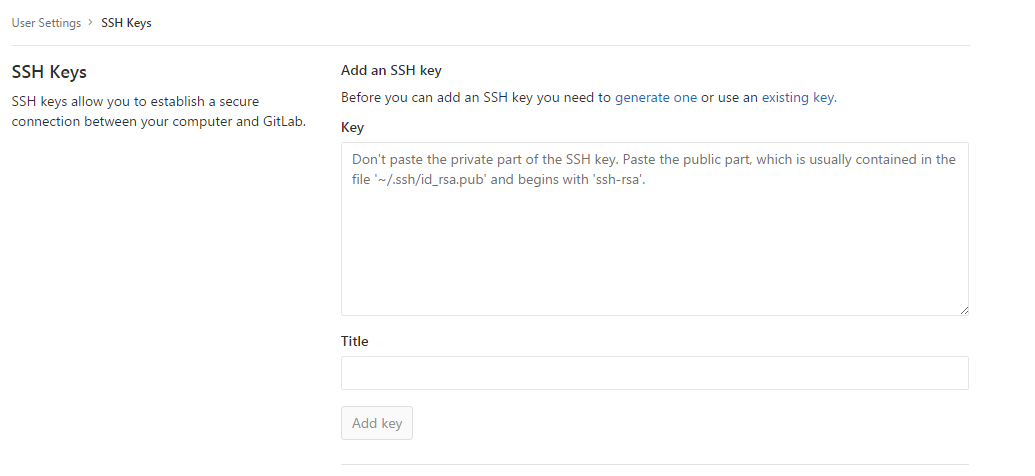
**Working with ssh for connecting with gitlab.**

First generate the ssh key in the local machine from where you want to upload the files.

Command - ssh-keygen

Open the file id\_rsa.pub from the location where it stored and copy the content and past it in the gitlab as per below steps.

From the top right corner in gitab go to -> settings -> ssh keys(right side) and add content of the public key.



**Backup and restore the git data.**

**Backup**

Run the below command which will take a back in the form of .tar and place it in /vat/opt/gitlab/backups

*gitlab-rake gitlab:backup:create*

tar – contains all the repo + users.

Backup should be created in the /var/opt/gitlab/backups folder.

**Restore**

Stop the two services prior to restoring.

*Gitlab-ctl stop unicron*

*Gitlab-ctl stop sidekiq*

gitlab-rake gitlab:backup:restore BACKUP=1521884424\_2018\_03\_24\_10.5.3

note- Backup argument should contain the file name of the tar excluding gitlab-backup.tar in its name.

example :

