DigitalOcean

⌄　　　　☐⁺ Subscribe　　　☐ Share　　　☰ Contents ⌄

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.　✕

Enter your email address　　　　　　　　　Sign Up

Jbuntu 16.04

SCROLL TO TOP

♡

Updated January 24, 2018   ◉ 262.9k   [APPLICATIONS]   [GIT]   [UBUNTU]   [UBUNTU 16.04]

By: Justin Ellingwood

Not using **Ubuntu 16.04**? Choose a different version:

Introduction

GitLab CE, or Community Edition, is an open source application primarily used to host Git repositories, with additional development-related features like issue tracking. It is designed to be hosted using your own infrastructure, and provides flexibility in deploying as an internal repository store for your development team, publicly as a way to interface with users, or even open as a way for contributors to host their own projects.

The GitLab project makes it relatively straight forward to set up a GitLab instance on your own hardware with an easy installation mechanism. In this guide, we will cover how to install and configure GitLab on an Ubuntu 16.04 server.

Prerequisites

This tutorial will assume that you have access to a fresh Ubuntu 16.04 server. The published GitLab hardware requirements recommend using a server with:

- 2 cores

- 4GB of RAM

Although you may be able to get by with substituting some swap space for RAM, it is not recommended. For this guide we will assume

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

Enter your email address   Sign Up

In order to get started, you will need a non-root user with `sudo` access configured on the server. It is also a good idea to set up a basic firewall to provide an additional layer of security. You can follow the steps in our Ubuntu 16.04 initial server setup guide to get this setup.

When you have satisfied the above prerequisites, continue on to start the installation procedure.

Installing the Dependencies

Before we can install GitLab itself, it is important to install some of the software that it leverages during installation and on an ongoing basis. Fortunately, all of the required software can be easily installed from Ubuntu's default package repositories.

Since this is our first time using `apt` during this session, we can refresh the local package index and then install the dependencies by typing:

```
$ sudo apt-get update
$ sudo apt-get install ca-certificates curl openssh-server postfix
```

You will likely have some of this software installed already. For the `postfix` installation, select **Internet Site** when prompted. On the next screen, enter your server's domain name or IP address to configure how the system will send mail.

Installing GitLab

Now that the dependencies are in place, we can install GitLab itself. This is a straight forward process that leverages an installation script to configure your system with the GitLab repositories.

Move into the `/tmp` directory and then download the installation script:

script.deb.sh

Feel free to examine the downloaded script to ensure that you are comfortable with the actions that it will take. You can also find a hosted version of the script here:

```
$ less /tmp/script.deb.sh
```

Once you are satisfied with the safety of the script, run the installer:

```
$ sudo bash /tmp/script.deb.sh
```

The script will set up your server to use the GitLab maintained repositories. This lets you manage GitLab with the same package management tools you use for your other system packages. Once this is complete, you can install the actual GitLab application with `apt`:

```
$ sudo apt-get install gitlab-ce
```

This will install the necessary components on your system.

Adjusting the Firewall Rules

Before you configure GitLab, you will need to ensure that your firewall rules are permissive enough to allow web traffic. If you followed the guide linked in the prerequisites, you will have a `ufw` firewall enabled.

View the current status of your active firewall by typing:

```
$ sudo ufw status
```

```
Status: active

To                          Action       From
--                          ------       ----
OpenSSH                     ALLOW        Anywhere
OpenSSH (v6)                ALLOW        Anywhere (v6)
```

As you can see, the current rules allow SSH traffic through, but access to other services is restricted. Since GitLab is a web application, we should allow HTTP access in. If you have a domain name associated with your GitLab server, GitLab can also request and enable a free TLS/SSL certificate from the Let's Encrypt project to secure your installation. We'll want to allow HTTPS access as well in this case.

Since the protocol to port mapping for HTTP and HTTPS are available in the `/etc/services` file, we can allow that traffic in by name. If you didn't already have OpenSSH traffic enabled, you should allow that traffic now too:

```
$ sudo ufw allow http
$ sudo ufw allow https
$ sudo ufw allow OpenSSH
```

If you check the `ufw status` command again, you should see access configured to at least these two services:

```
$ sudo ufw status
```

Output

```
Status: active

To                          Action       From
--                          ------       ----
```

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

Enter your email address   Sign Up

```
443                          ALLOW       Anywhere
OpenSSH (v6)                 ALLOW       Anywhere (v6)
80 (v6)                      ALLOW       Anywhere (v6)
443 (v6)                     ALLOW       Anywhere (v6)
```

The above output indicates that the GitLab web interface will be accessible once we configure the application.

Editing the GitLab Configuration File

Before you can use the application, you need update one configuration file and run a reconfiguration command. First, open Gitlab's configuration file:

```
$ sudo nano /etc/gitlab/gitlab.rb
```

Near the top is the `external_url` configuration line. Update it to match your own domain or IP address. If you have a domain, change `http` to `https` so that GitLab will automatically redirect users to the site protected by the Let's Encrypt certificate we will be requesting.

/etc/gitlab/gitlab.rb

```
# If your GitLab server does not have a domain name, you will need to use an IP
# address instead of a domain and keep the protocol as `http`.
external_url 'https://yourdomain'
```

Next, if your GitLab server has a domain name, search the file for the `letsencrypt['enable']` setting. Uncomment the line and set it to `true`. This will tell GitLab to request a Let's Encrypt certificate for your GitLab domain and configure the application to serve traffic with it.

will know of any issues:

/etc/gitlab/gitlab.rb

```
letsencrypt['enable'] = true
letsencrypt['contact_emails'] = ['sammy@yourdomain.com']
```

Save and close the file. Now, run the following command to reconfigure Gitlab:

```
$ sudo gitlab-ctl reconfigure
```

This will initialize GitLab using information it can find about your server. This is a completely automated process, so you will not have to answer any prompts. If you enabled the Let's Encrypt integration, a certificate should be configured for your domain.

Performing Initial Configuration Through the Web Interface

Now that GitLab is running and access is permitted, we can perform some initial configuration of the application through the web interface.

Logging In for the First Time

Visit the domain name of your GitLab server in your web browser:

```
http://gitlab_domain_or_IP
```

If you enabled Let's Encrypt and used `https` in your `external_url`, you should be redirected to a secure HTTPS connection.

...dministrative account:

In the initial password prompt, supply and confirm a secure password for the administrative account. Click on the **Change your password** button when you are finished.

You will be redirected to the conventional GitLab login page:

Your password has been changed successfully.

GitLab Community Edition

Open source software to collaborate on code

Manage Git repositories with fine-grained access controls that keep your code secure. Perform code reviews and enhance collaboration with merge requests. Each project can also have an issue tracker and a wiki.

Sign in   **Register**

Username or email

root

Password

••••••••

☐ Remember me   Forgot your password?

Sign in

Didn't receive a confirmation email? Request a new one.

Explore   Help   About GitLab

Here, you can log in with the password you just set. The credentials are:

- Username: **root**
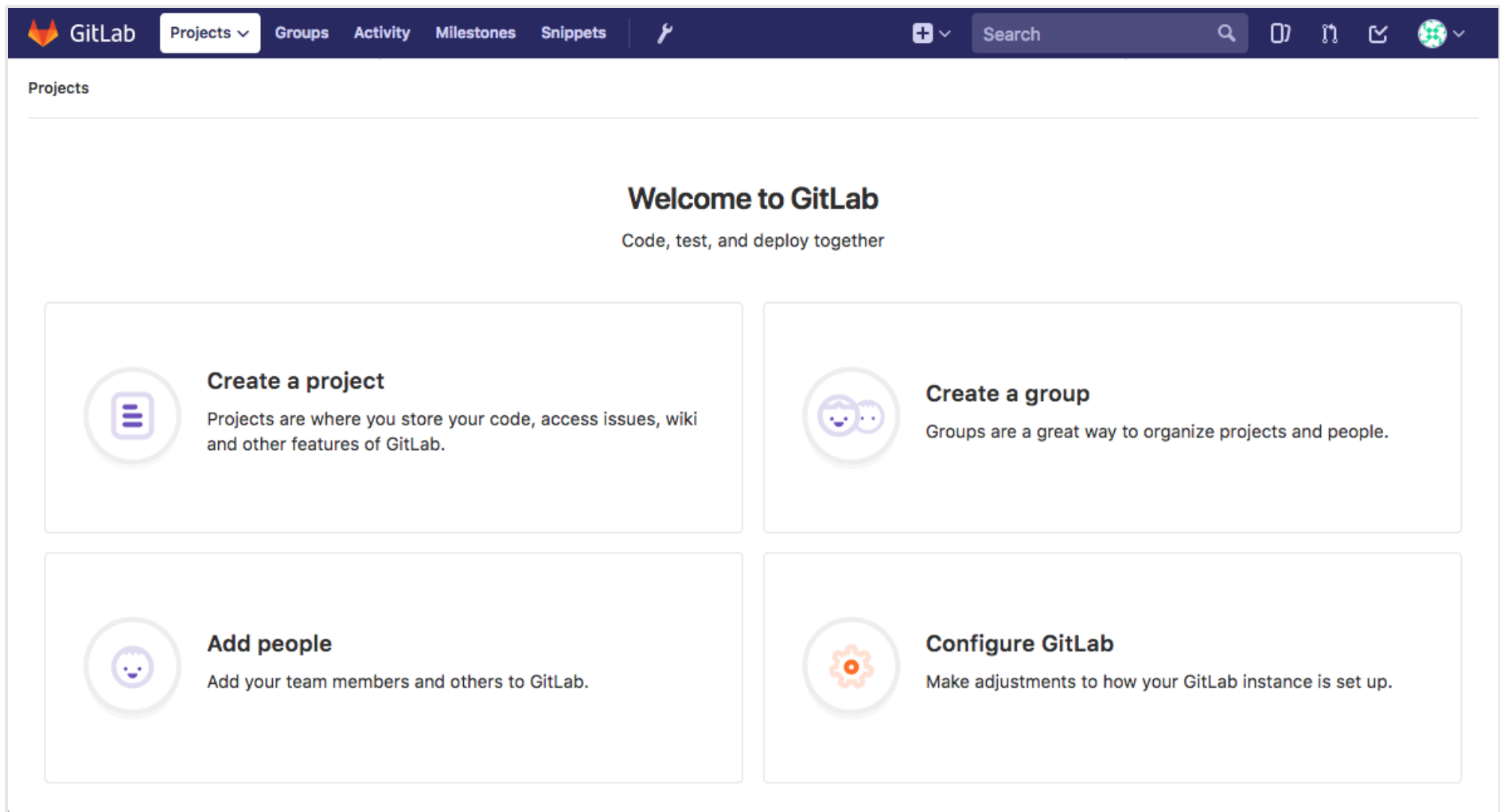
- Password: [the password you set]

Enter these values into the fields for existing users and click the **Sign in** button. You will be signed into the application and taken to a

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

Enter your email address   Sign Up

You can now make some simple changes to get GitLab set up the way you'd like.
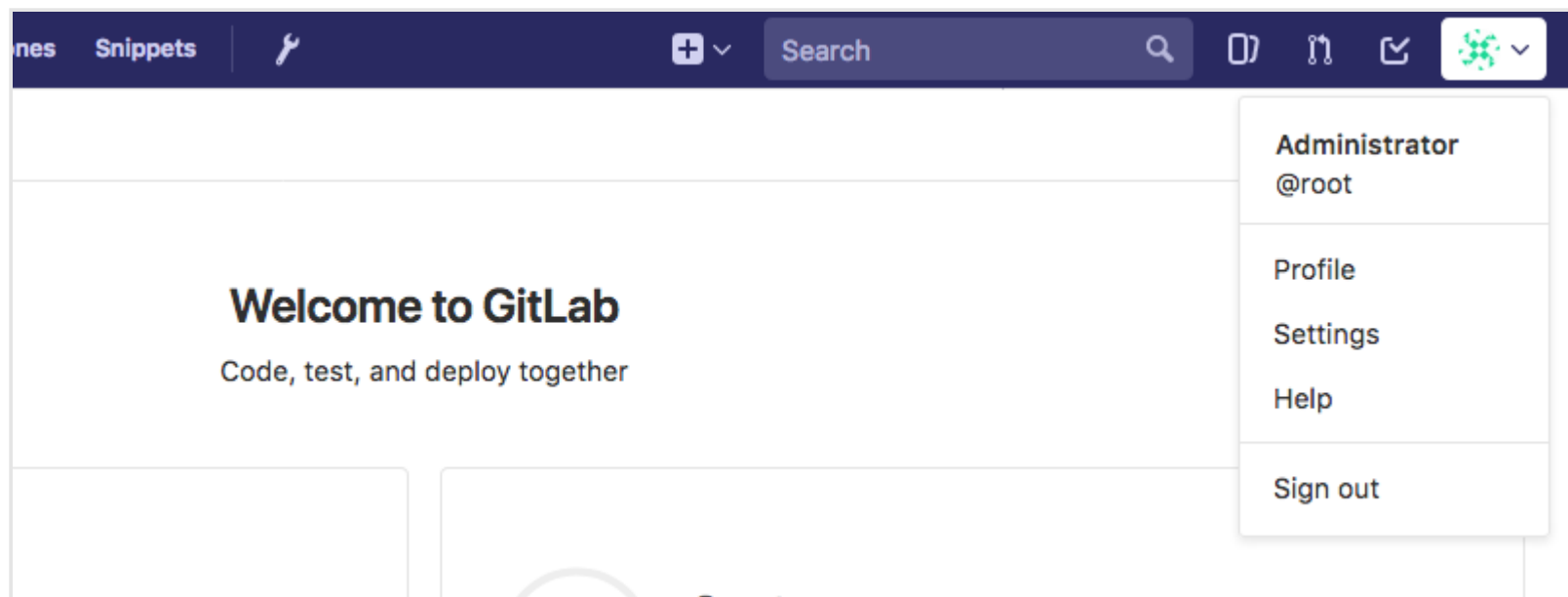
Adjusting your Profile Settings

One of the first things that you should do after a fresh installation is to get your profile into better shape. GitLab selects some reasonable defaults, but these are not usually appropriate once you start using the software.

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕   orner of the interface. In the drop down menu that
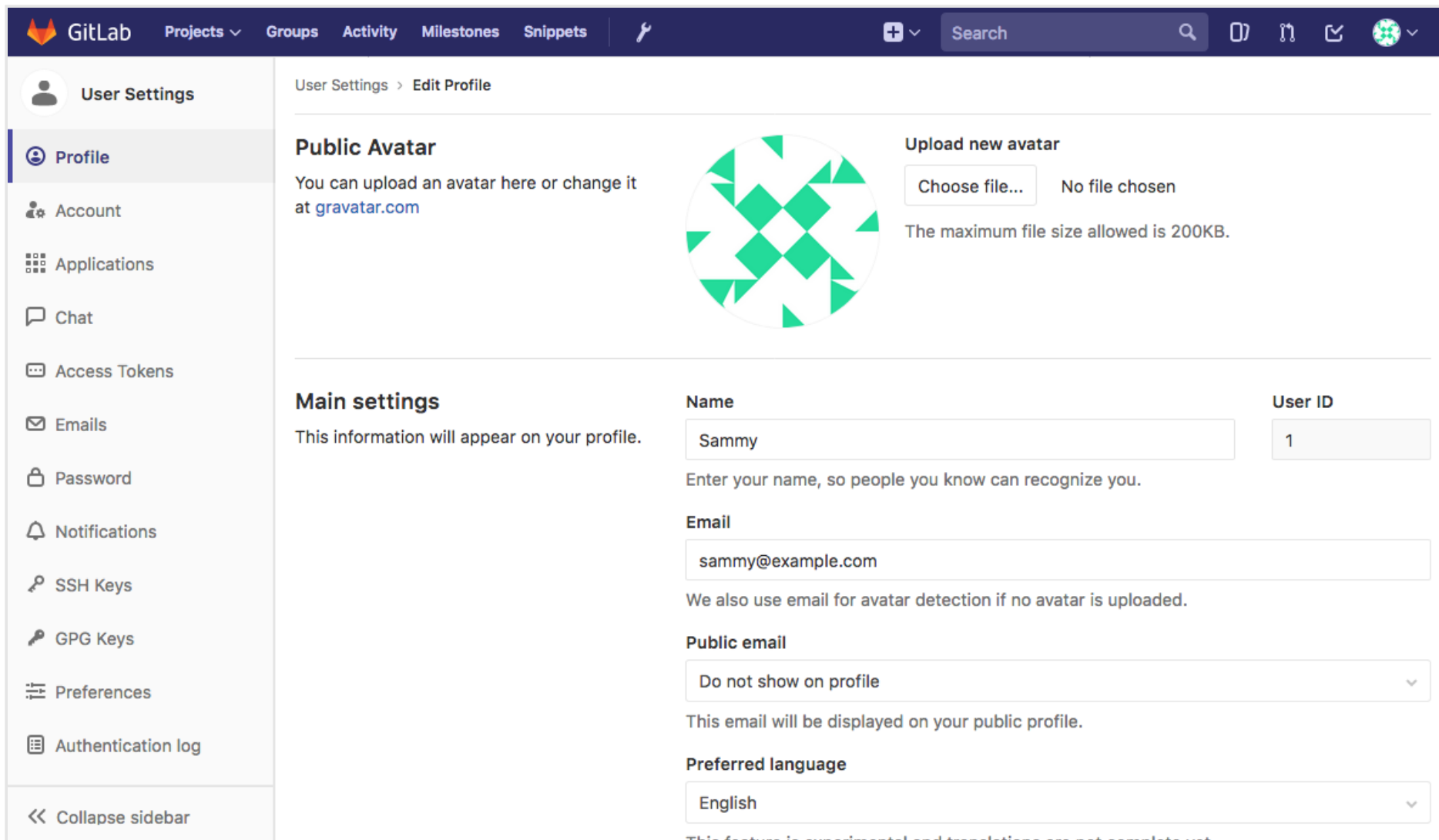
Enter your email address   Sign Up

You will be taken to the **Profile** section of your settings:

Adjust the **Name** and **Email** address from "Administrator" and "admin@example.com" to something more accurate. The name you select will be displayed to other users, while the email will be used for default avatar detection, notifications, Git actions through the interface, etc.

Click on the **Update Profile settings** button at the bottom when you are done:

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕
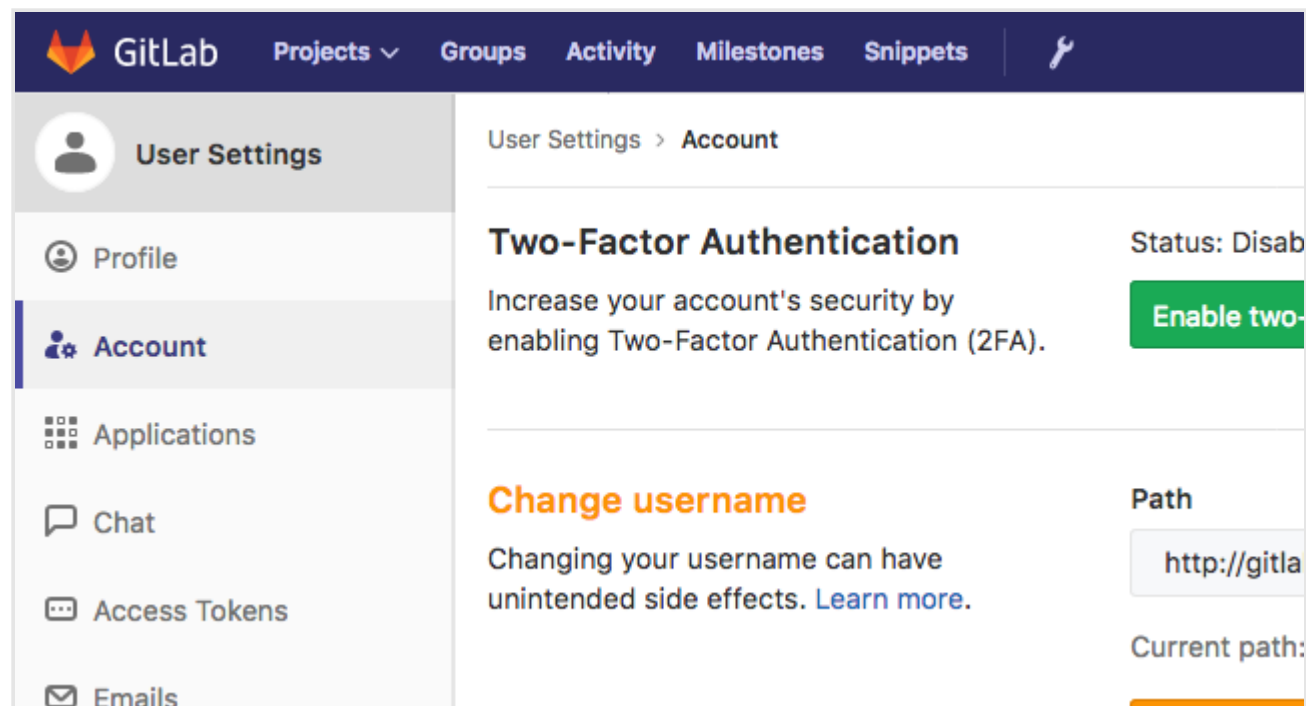
Enter your email address   Sign Up

Update profile settings

A confirmation email will be sent to the address you provided. Follow the instructions in the email to confirm your account so that you can begin using it with GitLab.

Changing Your Account Name

Next, click on the **Account** item in the left-hand menu bar:

Here, you can find your private API token or configure two-factor authentication. However, the functionality we are interested in at the moment is the **Change username** section.

By default, the first administrative account is given the name **root**. Since this is a known account name, its more secure to change this a different name. You will still have administrative privileges; the only thing that will change is the name:

Change username

Changing your username can have unintended side effects. Learn more.

Path

http://gitlab.boink.pro/   sammy

Current path: http://gitlab.boink.pro/root

Update username

Click on the **Update username** button to make the change:

Update username

Next time you log in to the GitLab, remember to use your new username.

Adding an SSH Key to your Account

In most cases, you will want to use SSH keys with Git to interact with your GitLab projects. To do this, you need to add your SSH public key to your GitLab account.

If you already have an SSH key pair created on your **local computer**, you can usually view the public key by typing:

```
local$ cat ~/.ssh/id_rsa.pub
```

Output

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDMuyMtMl6aWwqBCvQx7YXvZd7bCFVDsyln3yh5/8Pu23LW88VXfJgsBvhZZ9W0rPBGYyzE/TDzwwITvVQcKrwQr
```

Copy this text and head back to the Profile Settings page in GitLab's web interface.

If, instead, you get a message that looks like this, you do not yet have an SSH key pair configured on your machine:

Output

```
cat: /home/sammy/.ssh/id_rsa.pub: No such file or directory
```

If this is the case, you can create an SSH key pair by typing:

```
local$ ssh-keygen
```

Accept the defaults and optionally provide a password to secure the key locally:

Output

```
Generating public/private rsa key pair.
Enter file in which to save the key (/home/sammy/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/sammy/.ssh/id_rsa.
Your public key has been saved in /home/sammy/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:I8v5/M5xOicZRZq/XRcSBNxTQV2BZszjlWaIHi5chc0 sammy@gitlab.docsthat.work
```

```
|         ..%o==B|
|           *.E =.|
|        . ++= B  |
|          ooo.o .  |
|       . S .o   . .|
|      . + .. .    o|
|        +   .o.o ..|
|       o .++o .   |
|         oo=+     |
+----[SHA256]-----+
```

Once you have this, you can display your public key as above by typing:

```
local$ cat ~/.ssh/id_rsa.pub
```

Output

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAAABAQDMuyMtMl6aWwqBCvQx7YXvZd7bCFVDsyln3yh5/8Pu23LW88VXfJgsBvhZZ9W0rPBGYyzE/TDzwwITvVQcKrwQr
```
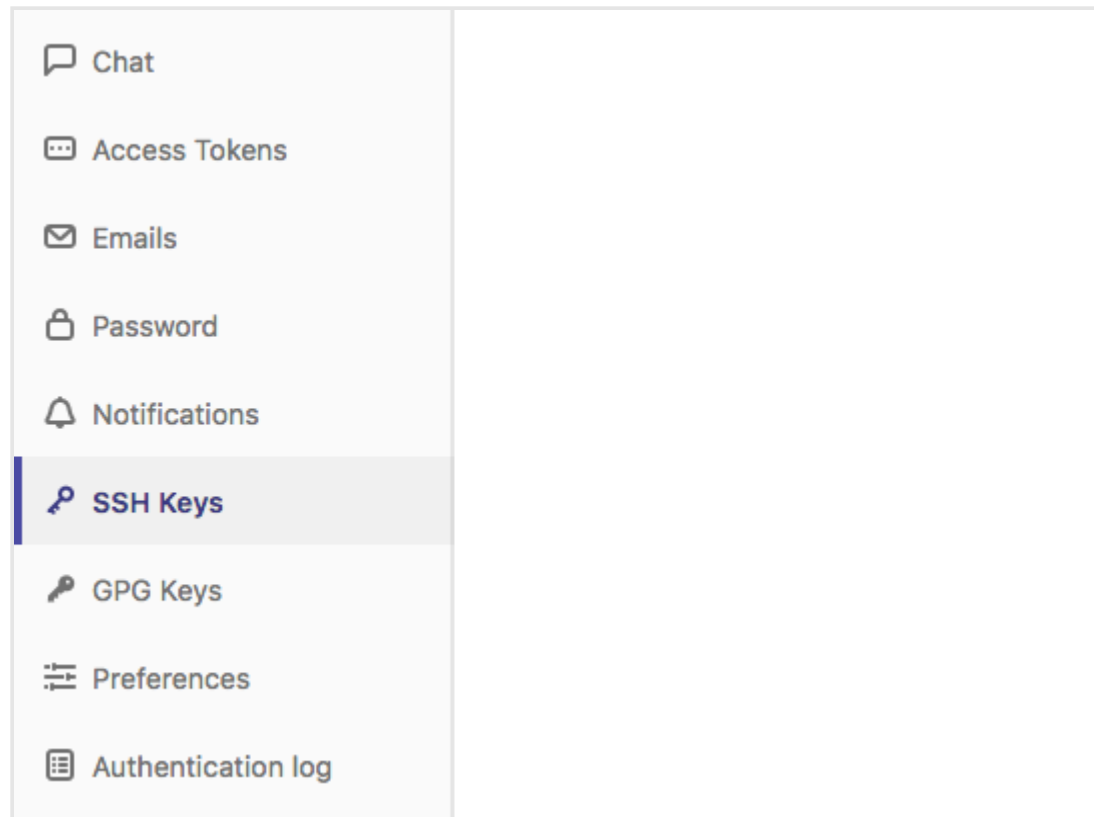
Copy the block of text that's displayed and head back to your Profile Settings in GitLab's web interface.

Click on the **SSH Keys** item in the left-hand menu:

In the provided space paste the public key you copied from your local machine. Give it a descriptive title, and click the **Add key** button:

SSH Keys

SSH keys allow you to establish a secure connection between your computer and GitLab.

Add an SSH key

Before you can add an SSH key you need to *generate it.*

Key

```
ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIGRduW5WBpZPyYA4MhD5lUb9/HzPPwKBQQwXWw15F6
OJ sammy@shark
```

Title

sammy@shark

Add key

You should now be able to manage your GitLab projects and repositories from your local machine without having to provide your GitLab account credentials.

Restricting or Disabling Public Sign-ups (Optional)

You may have noticed that it is possible for anyone to sign up for an account when you visit your GitLab instance's landing page. This may be what you want if you are looking to host public project. However, many times, more restrictive settings are desirable.
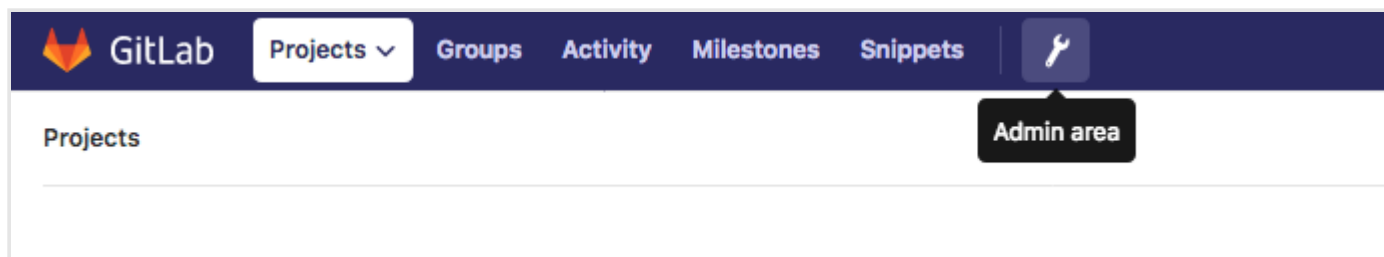
To begin, make your way to the administrative area by clicking on the **wrench icon** in the main menu bar at the top of the page:

On the page that follows, you can see an overview of your GitLab instance as a whole. To adjust the settings, click on the **Settings** item at the bottom of the left-hand menu.
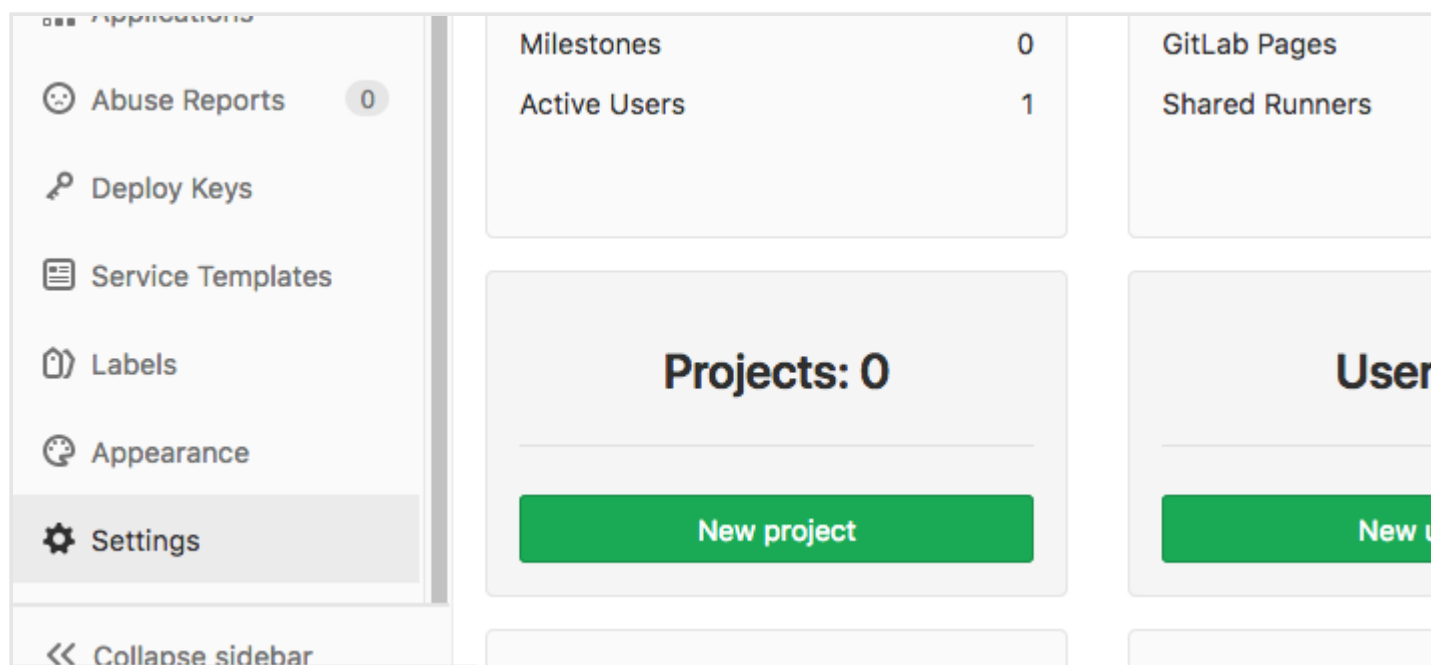
You will be taken to the global settings for your GitLab instance. Here, you can adjust a number of settings that affect whether new users can sign up and what their level of access will be.
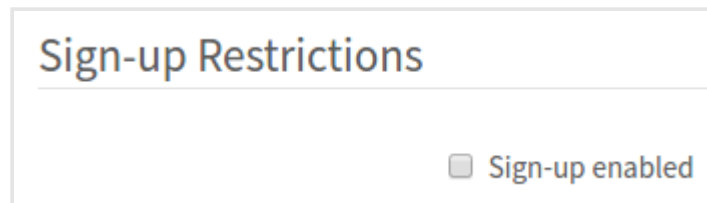
Disabling Sign-ups

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics. ✕ new users), scroll down to the **Sign-up**

Enter your email address   Sign Up

Deselect the **Sign-up enabled** check box:

Scroll down to the bottom and click on the **Save** button:

The sign-up section should now be removed from the GitLab landing page.

Restricting Sign-ups By Domain

If you are using GitLab as part of an organization that provides email addresses associated with a domain, you can restrict sign-ups by domain instead of completely disabling them.

In the **Sign-up Restrictions** section, first select the **Send confirmation email on sign-up** box only allow users to log in after they've confirmed their email.

Next, add your domain or domains to the **Whitelisted domains for sign-ups** box, one per line. You can use the asterisk "*" to specify wildcard domains:

Sign-up Restrictions

☑ Sign-up enabled

☑ Send confirmation email on sign-up

Whitelisted domains for   example.com
sign-ups   *.example.com

Scroll down to the bottom and click on the **Save** button:

Save

The sign-up section should now be removed from the GitLab landing page.

Restricting Project Creation
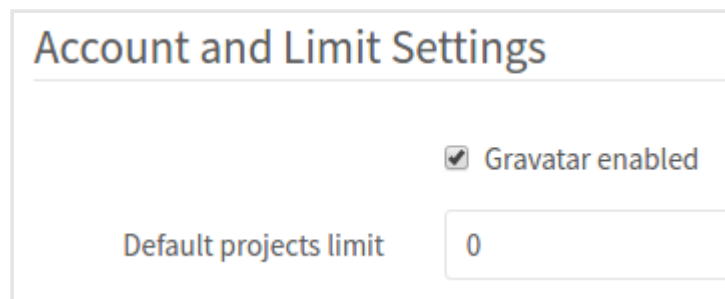
By default, new users can create up to 10 projects. If you wish to allow new users from the outside for visibility and participation, but want to restrict their access to creating new projects, you can do so in the **Account and Limit Settings** section.

Inside, you can change the **Default projects limit** to 0 to completely disable new users from creating projects:

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

Enter your email address   Sign Up
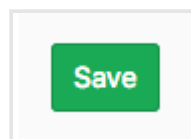
Account and Limit Settings

☑ Gravatar enabled

Default projects limit   0

New users can still be added to projects manually and will have access to internal or public projects created by other users.

Scroll down to the bottom and click on the **Save** button:

Save
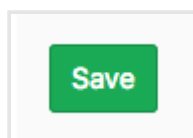
New users will now be able to create accounts, but unable to create projects.

Creating a Cron Job To Automatically Renew Let's Encrypt Certificates

By design, Let's Encrypt certificates are only valid for 90 days. If you enabled Let's Encrypt for your GitLab domain earlier, you will need to ensure that your certificates are renewed on a regular basis to avoid service interruptions. GitLab provides the `gitlab-ctl renew-le-certs` command to request new certificates when your current assets approach their expiration.

To automate this process, we can create a cron job to automatically run this command on a regular basis. The command will only renew the certificate when it's close to expiring, so we can safely run it regularly.

To begin, create and open a file at `/etc/cron.daily/gitlab-le` in your text editor:

Inside, paste the following script:

/etc/cron.daily/gitlab-le

```
#!/bin/bash

set -e

/usr/bin/gitlab-ctl renew-le-certs > /dev/null
```

Save and close the file when you are finished.

Mark the file as executable by typing:

```
$ sudo chmod +x /etc/cron.daily/gitlab-le
```

Now, GitLab should automatically check each day if its Let's Encrypt certificate needs to be renewed. If it does, the command will renew the certificate automatically.

Conclusion

You should now have a working GitLab instance hosted on your own server. You can begin to import or create new projects and configure the appropriate level of access for your team. GitLab is regularly adding features and making updates to their platform, so be sure to check out the project's home page to stay up-to-date on any improvements or important notices.

By: Justin Ellingwood   ♡ Upvote (32)   ⬆ Subscribe   ⬆ Share

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

Enter your email address   Sign Up

Open Source Presentation Grants

Receive free infrastructure credits to power your next tech talk or live demo.

LEARN MORE

Related Tutorials

How To Build a Node.js Application with Docker

How To Install and Set Up Manifold Scholar

How To Send Web Push Notifications from Django Applications

How To Build a Modern Web Application to Manage Customer Information with Django and React o...

How To Use Git: A Reference Guide

32 Comments

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

| Enter your email address | Sign Up |

Leave a comment...

Log In to Comment

Shiretoko *September 3, 2016*

0 Hi, this is cool tutorials and easy for me to learn (I'm a beginning)
but I have a problem about the domain etc, the url path of avatar
it show my droplets name like "newgit-ubuntu-1gb-sgp1-01"
so in my avatar the url like this ==> "http://newgit-ubuntu-1gb-sgp1-01/uploads/user/avatar/1/avatar.png"
where can I edit the config file I can change "newgit-ubuntu-1gb-sgp1-01" to the right domain .
Please help ,thank you (sorry about my english is not well)

jellingwood MOD   *September 6, 2016*

3 @Shiretoko: You should be able to edit the `/etc/gitlab/gitlab.rb` file to make the change. Open the file in your editor:

```
$ sudo nano /etc/gitlab/gitlab.rb
```

Find the `external_url` setting and modify it to point to your correct domain:

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

Enter your email address   Sign Up

```
. . .
external_url 'http://your_domain_here'
. . .
```

Save and close the file.

Afterwards, reconfigure GitLab so that your domain name will propagate to all of the configuration files that GitLab manages internally:

```
$ sudo gitlab-ctl reconfigure
```

You should hopefully be able to access GitLab from your domain name at this point.

Hope that helps!

Shiretoko *September 6, 2016*

0 It working now ,Thank you so much :D

---

robertlvilvert *October 28, 2016*

0 I had the same problem with avatars.
But i just changed the host parameter on file /opt/gitlab/embedded/service/gitlab-rails/config/gitlab.yml

Like this,

host : your-machine-name
TO
host : your-ip-server or DNS

thedude *March 4, 2017*

0 Don't mess with the files under `/opt/gitlab/` , your changes will be deleted after upgrade. Use `/etc/gitlab/gitlab.rb` to persist your changes.

yaeykay *November 6, 2016*

0 Thank you for this new awesome tutorial @jellingwood. Is there a tutorial like this here in DO about installation and configuration of GitLab Runner for GitLab CI?

thedude *November 8, 2016*

0 I don't think there is one yet, but you can use the one hosted on our blog https://about.gitlab.com/2016/04/19/how-to-set-up-gitlab-runner-on-digitalocean/ :)

It might be a good idea to also have this on DO.

yaeykay *December 17, 2016*

0 @thedude dude Yeah. I already read the blog and still stuck with SSH Permission Denied.

Ram1976 *November 9, 2016*

0 @@jellingwood I have followed the steps in the above mentioned blog assuming I do not need any extra script. The only diff is I had my user name and group name as gitadmin. I tried a text only browser links http://localhost/ . I did ensure that port 80 and the ssh is allowed in ufw. But I am still seeing apache http server default home page instead of landing page of gitlab. Can you tell me what am i missing here.

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics. ✕

Enter your email address | Sign Up

Is it recommended to install all that on a already running webserver with 2 php projects? The workload is not too much (400+ Visits a day)

0

---

thedude *March 4, 2017*

o It depends how beefy your droplet is, check GitLab's requirements here https://docs.gitlab.com/ce/install/requirements.html#hardware-requirements.

Also, you'll need to configure GitLab to use an external web server as it uses a bundled Nginx by default https://docs.gitlab.com/omnibus/settings/nginx.html#using-a-non-bundled-web-server. If you don't do that, your other sites will probably be inaccessible.

---

mroceanms *January 9, 2017*

o i have a problem with the file upload of avatars...it fails every round.

no, there are no firewall restrictions ;-) Firewall ist disabled...

Any idea?

MartynKeigher *January 16, 2017*

o I added a line to my `/etc/hosts` file to workaround this 'out-of-the-box' known issue for now...

Entry looks like this:

```
/etc/hosts

127.0.0.1        localhost
127.0.1.1        gitlab.lab.mydomain.com     gitlab
```

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

Enter your email address   Sign Up

Basically: The local IP of the server running gitlab has to point to the externally facing FQDN of itself.

EDIT: Some more info too... be sure to make the edit mentioned above in QnA section regarding the FQDN in the `/etc/gitlab/gitlab.rb` (line 11!) ;p

ihavenolife *February 4, 2017*

0 Hi,

Great tutorial! Although I have installed using preconfigured image this still is very good and helps to understand the underpinings.

I have a question regarding if it is possible to run side by side web server apache/nginx with GitLab.

I was reading through the manual on GitLab page and it looks like it comes bundled with nginx. But I cannot locate it anywhere. So I'm curious if I'm missing something?

thedude *March 4, 2017*

0 You are correct. If you want to use an external web server, check these docs
https://docs.gitlab.com/omnibus/settings/nginx.html#using-a-non-bundled-web-server.

pmamalster *March 30, 2017*

0 [Help] I am getting this error

Error executing action `run` on resource 'execute[clear the gitlab-rails cache]

after which my installation fails

Running handlers:

Running handlers complete

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

Enter your email address   Sign Up

earnshae *August 17, 2017*

You need to add an apache2 restart to the above text to cause apache to serve gitlab.

```
sudo service restart apache2
```

sheikh303 *September 26, 2017*

Hi,

I just created an droplet of 4G ram and 2 core CPU. Followed the instructions as given here.
But right after installing when I browse the GITLab through public IP getting 502 error.

sheikh303 *September 29, 2017*

Looking forward to get a fix here soon. thanks!

jellingwood MOD *September 29, 2017*

@sheikh303 Sorry to hear that. Unfortunately, I'm not able to reproduce the problem you're describing. I just tried to install twice using these instructions and was able to access the web interface both times. It's difficult to say what could be causing. I would recommend checking that all of the backend services are up and running by typing:

```
$ sudo gitlab-ctl status
```

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

Enter your email address   Sign Up

```
$ sudo gitlab-ctl start service-name
```

You can also try restarting all components by typing:

```
$ sudo gitlab-ctl restart
```

It may also be a good idea to check to make sure you can reconfigure the application correctly by typing:

```
$ sudo gitlab-ctl reconfigure
```

If all of these work correctly, you might want to start over from scratch to see if there might have been a failure elsewhere. Sorry I can provide any more specific help. I hope you're able to figure it out.

---

wolfi *September 29, 2017*

0 hi, i followed the steps in this tutorial but sadly i cant see the gitlab webinterface
   there is still the apache2 ubuntu screen
   what did i missed?

---

kurisu *November 24, 2017*

0 Well, color me frustrated. On a fairly new, fairly empty droplet, I followed these steps. The Chef client failed. And now I have some runaway git processes that I can't kill, and can't uninstall (because there's no memory left to allocate for even the simplest of operations).
   Not saying it's for lack of good info in this article. But wondering if the script is somehow out of date now, and causing an issue. Anyone had any luck with this in late 2017?

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

Enter your email address   Sign Up

⁰ My bad. I didn't have enough RAM. Pure & simple.

---

angelovillasant *March 5, 2018*

₀ Does this work for Ubuntu 14.04?

---

lostideaslab *March 7, 2018*

₀ Just a budget question, can you install this on a droplet that running some low RAM & Space Consuming website, alongside them? or you need a blank server?

---

jellingwood MOD   *March 9, 2018*

₀ @lostideaslab You probably want to avoid installing GitLab on a server running active websites. For one thing, GitLab uses port 443 and port 80 for its own purposes, using an internally managed web server. That's going to conflict with your web server software, meaning you'd have to run either GitLab or your web sites on a non-standard port. Over all, it's probably best to dedicate a machine to running your GitLab server to avoid conflicts and isolate your systems.

---

bluemagma *March 12, 2018*

₀ Excellent documentation work, thank you so very much!

---

sebvarga *April 4, 2018*

₀ I wonder if anyone has problems with the config file. There's no file /etc/gitlab/gitlab.rb

And when I create it and add the above suggested code for the domain, it has no effect. I require to change the Domains and URLs that are provided for the new user mails, but I can't figure them out

Can someone clarify?

---

abhimarkan *June 5, 2018*

Hi There,

Just to confirm once I push anything to my repo (running on my droplet) all my code base will be private and will not be able available on public repos?

Thanks

---

aparkfa98075270 *June 20, 2018*

If you set HTTPS and get the Cert error.
===
Error executing action `create` on resource 'acme_certificate[staging]'
===

Bug Report
https://gitlab.com/gitlab-org/gitlab-ce/issues/43719

nginx['redirect*http*to*https*port'] = 80

---

iscdeyanira *July 6, 2018*

Hi.

I want to say thank you for the information, It was very helpful to continue with my project.

:)

[sanslogique](#) *August 23, 2018*

0 This was the most complete tutorial I found on the internet. Congratulations!

I followed all the steps and I was able to install normally, but I'm not receiving the confirmation email to login. Even though I have enabled the function, I can not receive the email to log in. I am a beginner and I even searched the internet, but I did not find any solution to my problem.

Could you give me a light?

Thanks!

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

Enter your email address   Sign Up

Community   Tutorials   Questions   Projects   Tags   Newsletter   RSS 🔊

Distros & One-Click Apps   Terms, Privacy, & Copyright   Security   Report a Bug   Write for DOnations   Shop

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

Enter your email address   Sign Up