

Як налаштувати наш  
власний сервер git за  
допомогою gitlab

# Вступ

Якщо ви користувач Git, ви знаєте, що наявність локальних сховищ, до яких можна отримати доступ через локальну мережу (або зовнішню глобальну мережу), є ключовим елементом процесу розробки. Звичайно, ви можете вибрати GitHub, але це позбавляє можливості локального розміщення. Отже, якщо ви хочете розмістити власні сховища, куди вам звернутися?

Одним словом, GitLab.

GitLab дозволяє розмістити локальне сховище Git, до якого можна отримати доступ або з вашої локальної мережі, або (якщо у вас є доступна загальнодоступна IP-адреса) за межами вашої компанії. GitLab досить простий у встановленні та неймовірно простий у використанні. Я збираюся провести вас через процес встановлення GitLab на Ubuntu Server 16.04. Я припускаю, що у вас уже запущено сервер Ubuntu і ви маєте доступ до облікового запису з правами sudo.

# Встановлення

Щоб встановити GitLab, вам знадобиться сервер із принаймні двома ядрами та 4 ГБ оперативної пам'яті. Якщо ви використовуєте це на віртуальній машині, переконайтеся, що віртуальна машина перевищує ці ресурси.

Перше, що вам потрібно зробити, це запустити оновлення або оновлення. Зауважте, що у разі оновлення ядра знадобиться перезавантаження. Якщо це виробнича машина, не забудьте запустити цей процес у неробочий час.

Відкрийте вікно терміналу на сервері та введіть команди:

```
sudo apt update  
sudo apt upgrade
```

# Встановлення

Після виконання цих команд ви готові встановити GitLab (за умови, що вам не потрібно перезавантажувати сервер).

Перше, що необхідно встановити, це необхідні залежності. Це можна вирішити за допомогою такої команди:

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

Під час інсталяції вище вас запитують, як налаштувати Postfix. Виберіть Інтернет-сайт, а потім введіть домен або IP-адресу сервера. Якщо ваші користувачі знайомі з Linux, ви завжди можете вибрати лише локальну конфігурацію Postfix, знаючи, що всі користувачі повинні використовувати команду mail на сервері, щоб перевірити, чи мають вони якусь пошту, доставлену GitLab. Якщо ви підете цим шляхом, вам потрібно буде встановити пакет mailutils, наприклад:

```
sudo apt install mailutils
```

# Встановлення

З готовими залежностями ми повинні встановити необхідний репозиторій за допомогою таких

```
curl -LO https://packages.gitlab.com/install/repositories/gitlab/gitlab-ce/script.deb.sh  
cd /tmp  
sudo bash /tmp/script.deb.sh
```

Нарешті, встановіть GitLab за допомогою команди:

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

# Налаштування GitLab

Існує лише один файл для редагування конфігурації. Відкрийте файл для редагування командою `sudo nano /etc/gitlab/gitlab.rb`. У цьому файлі ви шукаєте такий рядок:

```
external_url 'https://yourdomain'
```

Змініть це, щоб відобразити домен вашого сервера або IP-адресу. Якщо ви використовуєте IP-адресу, не забудьте відмовитися від `https` на користь `http`.

Якщо ви налаштуєте GitLab на використання домену, вам потрібно буде ввімкнути SSL. Для цього знайдіть наступні два рядки (навколо рядка 1519):

```
# letsencrypt['enable'] = false  
# letsencrypt['contact_emails'] = [ ]
```

# Налаштування GitLab

Розкоментуйте ці рядки (видаліть #), а потім змініть false на true та введіть дійсну адресу електронної пошти всередині порожніх символів [ ].

Збережіть і закрийте цей файл. Переналаштуйте GitLab за допомогою команди:

```
sudo gitlab-ctl reconfigure
```

# Налаштування GitLab

Після переналаштування відкрийте браузер і вкажіть у ньому `https://DOMAIN` або `http://IP` (де DOMAIN – це домен сервера, а IP – це IP-адреса сервера). Вам буде запропоновано змінити пароль облікового запису адміністратора (Малюнок А).

Please create a password for your new account.

## 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.

### Change your password

New password

.....

Confirm new password

Change your password

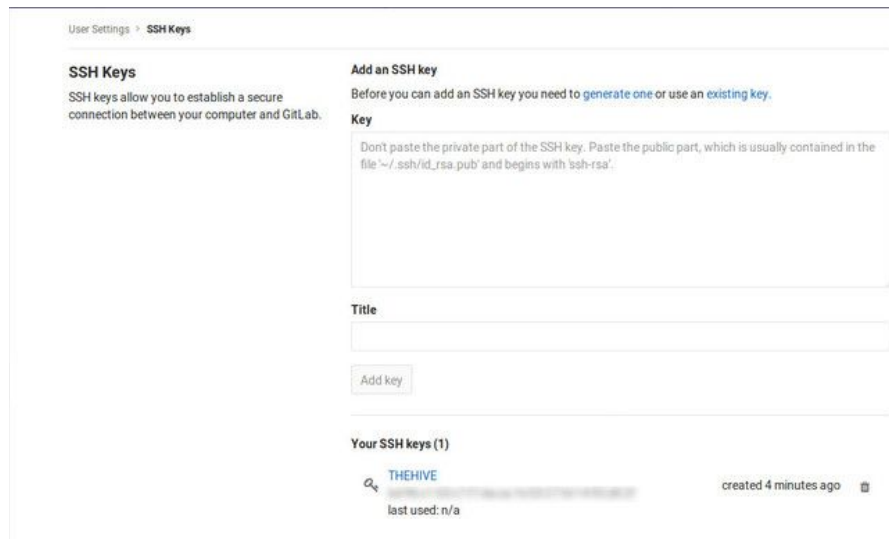
Didn't receive a confirmation email? [Request a new one](#)

Already have login and password? [Sign in](#)

# Налаштування GitLab: Додавання ключів ssh

Щоб надсилати або витягувати свої проекти, ви повинні додати ssh-ключі облікового запису віддаленого комп'ютера до свого нового облікового запису GitLab. Щоб зробити це, знайдіть ssh pubkey на віддаленій машині — у Linux це можна зробити, виконавши команду `cat ~/.ssh/id_rsa.pub` і скопіювавши вихідні дані. Повернувшись до свого облікового запису GitLab, клацніть спадне меню «Профіль» у верхньому правому куті та натисніть «Налаштування». У вікні «Параметри» клацніть «Ключі SSH» (Малюнок В).

Скопіюйте вміст SSH pubkey у розділі «Ключ», дайте ключу назву та натисніть «Додати ключ». З доданим відкритим ключем тепер ви зможете надсилати та витягувати зміни на сервер GitLab.



User Settings > SSH Keys

### 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 one](#) or use an [existing key](#).



**Key**

Don't paste the private part of the SSH key. Paste the public part, which is usually contained in the file `~/.ssh/id_rsa.pub` and begins with `'ssh-rsa'`.

**Title**

---

**Your SSH keys (1)**

 <b>THEHIVE</b>	created 4 minutes ago 
last used: n/a	

# Питання та відповіді