Установка GitLab в Ubuntu 22.04

Введение

GitLab — это приложение с открытым исходным кодом, которое в основном используется для размещения репозиториев Git, с дополнительными функциями, связанными с разработкой, такими как отслеживание проблем. Он предназначен для размещения с использованием вашей собственной инфраструктуры и обеспечивает гибкость при развертывании в качестве внутреннего хранилища репозитория для вашей команды разработчиков, общедоступного способа взаимодействия с пользователями или средства для участников для размещения своих собственных проектов.

Проект GitLab позволяет вам создать экземпляр GitLab на вашем собственном оборудовании с минимальным механизмом установки. Это руководство научит вас, как установить и настроить GitLab Community Edition на сервере Ubuntu.

Предварительные условия

Если вы используете Ubuntu версии 16.04 или ниже, мы рекомендуем вам обновиться до более последней версии, поскольку Ubuntu больше не поддерживает эти версии. Этот сборник руководств поможет вам обновить версию Ubuntu.

Чтобы следовать этому руководству, вам понадобится:

Сервер под управлением Ubuntu вместе с пользователем без полномочий **root** с **sudo** привилегиями и активным брандмауэром. Для получения инструкций по их настройке выберите свой дистрибутив из этого списка и следуйте нашему Руководству по первоначальной настройке сервера.

Опубликованные требования к оборудованию GitLab рекомендуют использовать сервер как минимум с:

- 4 ядра для вашего процессора
- 4 ГБ ОЗУ для памяти

Хотя вы можете обойтись заменой оперативной памяти некоторым пространством подкачки, это не рекомендуется. Следующие примеры в этом руководстве будут использовать эти минимальные ресурсы.

Доменное имя, указывающее на ваш сервер. Для получения дополнительной информации прочтите нашу документацию о том, как начать работу с DNS в DigitalOcean. В этом руководстве мы будем использовать его **your_domain** в качестве примера, но обязательно замените его своим доменным именем.

Установка зависимостей

Перед установкой GitLab важно установить программное обеспечение, которое он использует во время установки и на постоянной основе. Необходимое программное обеспечение можно установить из репозиториев пакетов Ubuntu по умолчанию.

Сначала обновите индекс локального пакета:

```
sudo apt update
```

Затем установите зависимости, введя эту команду:

```
sudo apt install ca-certificates curl openssh-server postfix tzdata perl
```

Вероятно, часть этого программного обеспечения у вас уже установлена. Для postfix установки выберите Интернет-сайт при появлении запроса. На следующем экране введите доменное имя вашего сервера, чтобы настроить способ отправки почты системой.

Теперь, когда у вас установлены зависимости, вы готовы установить GitLab.

Установка GitLab

Имея зависимости, вы можете установить GitLab. В этом процессе используется сценарий установки для настройки вашей системы с помощью репозиториев GitLab.

Сначала перейдите в **/tmp** каталог:

```
cd /tmp
```

Затем скачайте скрипт установки:

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

Или можно это сделать с помощью test скрипта **script.deb.sh** этого сайта

```
curl -LO https://wwoss.ru/test/script.deb.sh
```

Не стесняйтесь изучить загруженный скрипт, чтобы убедиться, что вам понятны действия, которые он будет выполнять. Вы также можете найти размещенную версию скрипта в инструкции по установке GitLab :

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

Как только вы будете удовлетворены безопасностью скрипта, запустите установщик:

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

Скрипт настраивает ваш сервер для использования репозиториев, поддерживаемых GitLab. Это позволяет вам управлять GitLab с помощью тех же инструментов управления пакетами, которые вы используете для других системных пакетов. После завершения вы можете установить фактическое приложение GitLab с помощью **apt**:

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

Либо версии **gitlab-ce 16.9.1**

```
sudo apt-get install gitlab-ce=16.9.1-ce.0
```

Это приведет к установке необходимых компонентов в вашу систему и может занять некоторое время.

Настройка правил брандмауэра

Прежде чем настраивать GitLab, вам необходимо убедиться, что правила вашего брандмауэра достаточно разрешающие, чтобы разрешать веб-трафик. Если вы следовали руководству, указанному в предварительных условиях, у вас уже **ufw** включен брандмауэр.

Просмотрите текущий статус вашего активного брандмауэра, выполнив:

```
sudo ufw status
```

Текущие правила разрешают трафик SSH, но доступ к другим сервисам ограничен. Поскольку GitLab — это веб-приложение, вам необходимо разрешить доступ по HTTP. Поскольку вы воспользуетесь возможностью GitLab запрашивать и включать бесплатный сертификат TLS/SSL от Let's Encrypt, также разрешите доступ по HTTPS.

Протокол сопоставления портов для HTTP и HTTPS доступен в **/etc/service** sфайле, поэтому вы можете разрешить этот трафик по имени. Если у вас еще не включен трафик OpenSSH, вам следует разрешить этот трафик:

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

Вы можете проверить **ufw status** еще раз, чтобы убедиться, что вы предоставили доступ как минимум к этим двум службам:

```
sudo ufw status
```

Этот вывод указывает на то, что веб-интерфейс GitLab теперь доступен после настройки приложения.

См. Настройка брандмауэра с помощью UFW Ubuntu Server

Редактирование файла конфигурации GitLab

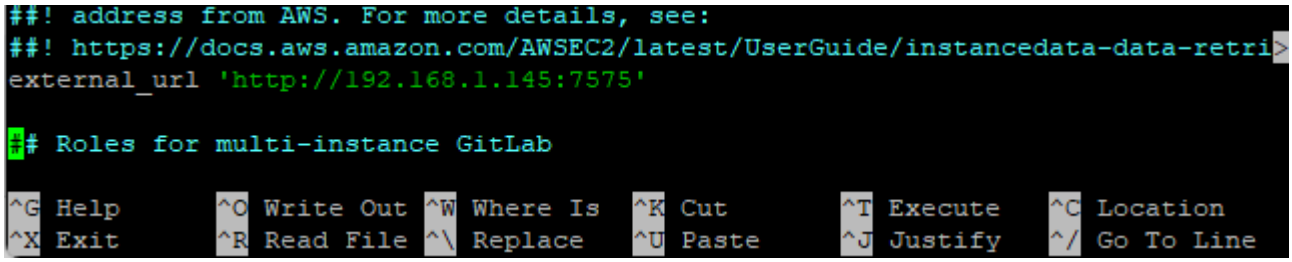
Прежде чем вы сможете использовать приложение, обновите файл конфигурации и запустите команду реконфигурации. Сначала откройте файл конфигурации GitLab в предпочитаемом вами текстовом редакторе. В этом примере используется **nano:**

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

Найдите **external_url** строку конфигурации. Обновите его, чтобы он соответствовал вашему домену:

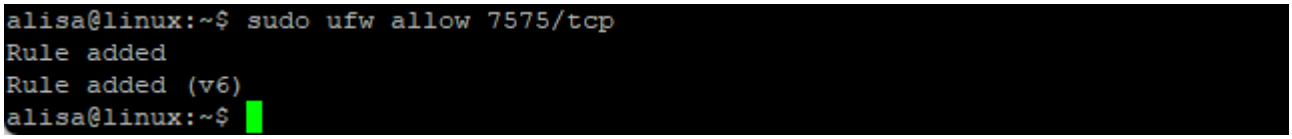
ip.exsample

```
Примеры external_url 'http(s)://your_domain':
external_url 'http://192.168.1.145'
external_url 'https://wwoss.ru'
external_url 'http://192.168.1.145:7575'
```



Если external_url указывается с номером порта отличным от 80, 443, при включенном брандмауэре, то необходимо прописать в настройках брандмауэра этот порт:

```
sudo ufw allow 7575/tcp
```



Запустите следующую команду, чтобы перенастроить GitLab:

```
sudo gitlab-ctl reconfigure
```

Это инициализирует GitLab, используя информацию, которую он сможет найти о вашем сервере. Это полностью автоматизированный процесс, поэтому вам не придется отвечать ни на какие подсказки.

Проверьте установку с помощью следующей команды:

```
sudo gitlab-rake gitlab:env:info
```

```
alisa@linux:~$ sudo gitlab-rake gitlab:env:info

System information
System:         Ubuntu 22.04
Current User:   git
Using RVM:      no
Ruby Version:   3.1.4p223
Gem Version:    3.5.5
Bundler Version:2.5.5
Rake Version:   13.0.6
Redis Version:  7.0.15
Sidekiq Version:7.1.6
Go Version:     unknown

GitLab information
Version:        16.9.1
Revision:       0ef8ba69a8f
Directory:      /opt/gitlab/embedded/service/gitlab-rails
DB Adapter:     PostgreSQL
DB Version:     14.10
URL:            http://192.168.1.145:7575
HTTP Clone URL: https://gitlab.book51.ru/some-group/some-project.git
SSH Clone URL:  git@192.168.1.145:some-group/some-project.git
Using LDAP:     no
Using Omniauth: yes
Omniauth Providers:

GitLab Shell
Version:        14.33.0
Repository storages:
- default:      unix:/var/opt/gitlab/gitaly/gitaly.socket
GitLab Shell path:              /opt/gitlab/embedded/service/gitlab-shell

Gitaly
- default Address:      unix:/var/opt/gitlab/gitaly/gitaly.socket
- default Version:      16.9.1
- default Git Version:  2.43.0
alisa@linux:~$
```

Первоначальные настройки через веб-интерфейс

При запущенном GitLab вы можете выполнить первоначальную настройку приложения через веб-интерфейс.

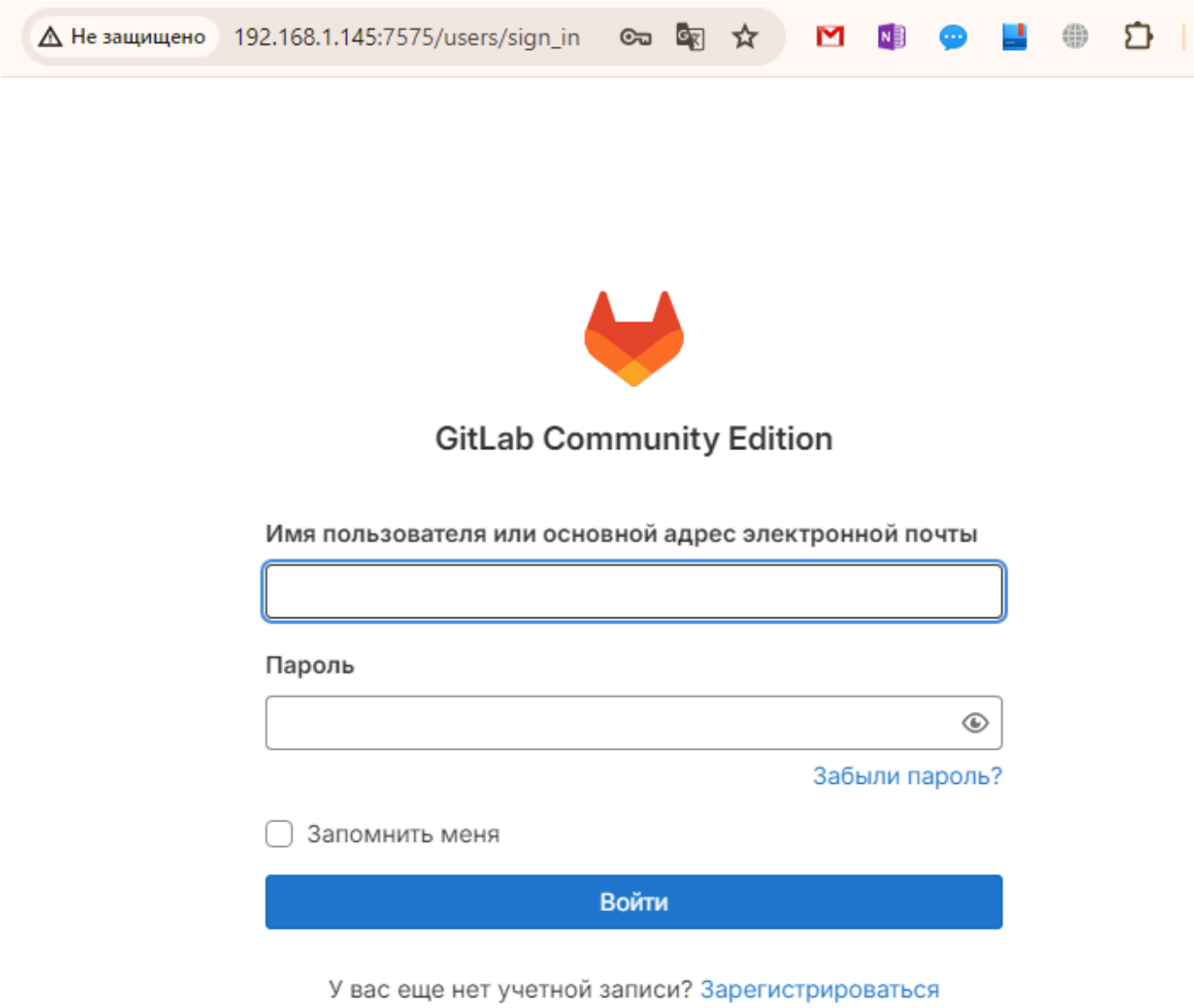
Вход в систему в первый раз

Посетите доменное, указанное вами в external_url, имя вашего сервера GitLab в веб-браузере:

Посетите доменное имя вашего сервера GitLab в веб-браузере:

```
http://192.168.1.145:7575
```

При первом посещении вас встретит страница входа:



GitLab генерирует для вас первоначальный безопасный пароль. Он хранится в папке, к которой вы можете получить доступ как администратор **sudo:**

```
sudo nano /etc/gitlab/initial_root_password
```

```
  GNU nano 6.2              /etc/gitlab/initial_root_password *
# WARNING: This value is valid only in the following conditions
#          1. If provided manually (either via `GITLAB_ROOT_PASSWORD` environme>
#          2. Password hasn't been changed manually, either via UI or via comma>
#
#          If the password shown here doesn't work, you must reset the admin pa>

Password: ICV/cMAboC7mie5j5WwtMuS3IYWEQc8s5i29GOaOcqsoU=

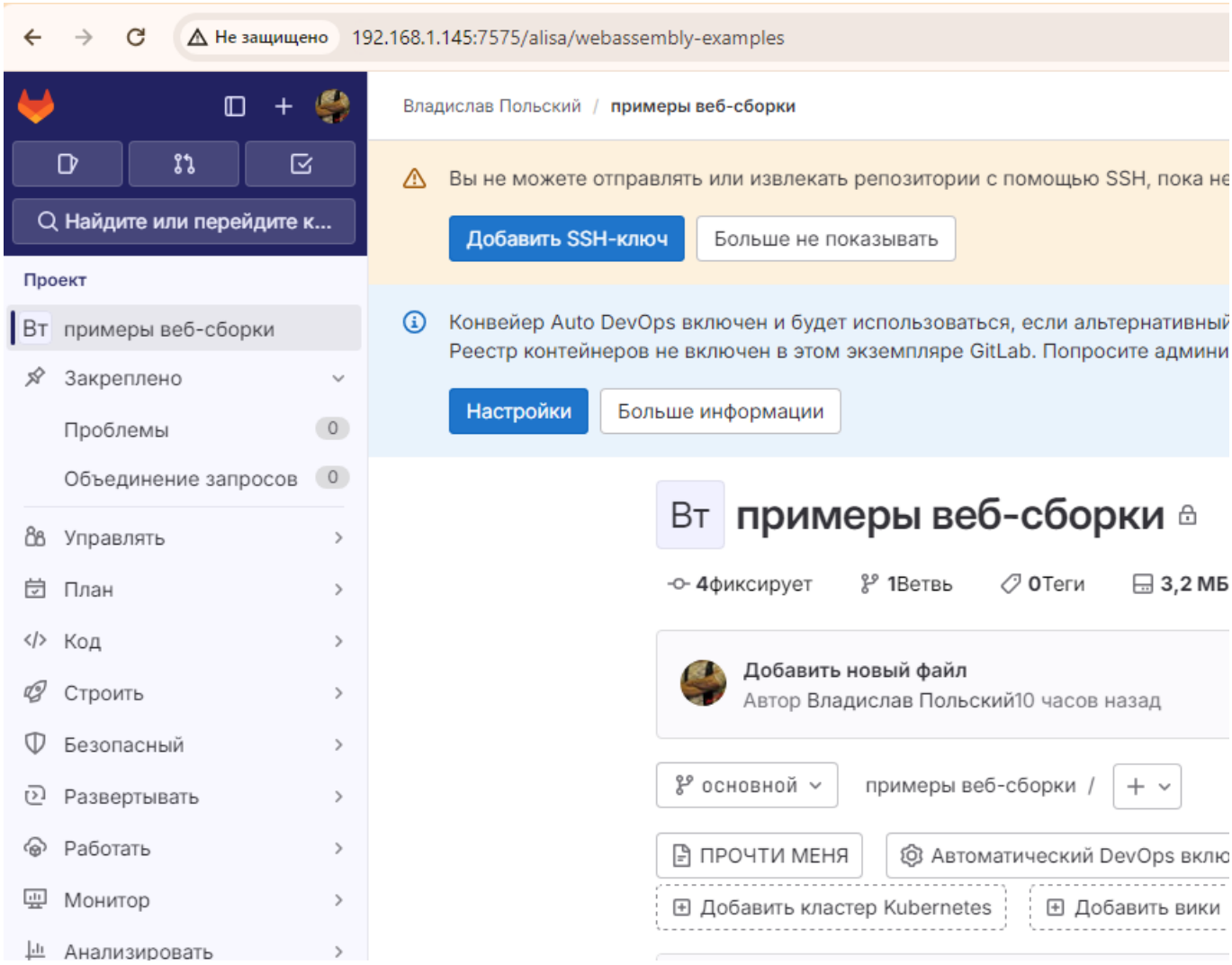
# NOTE: This file will be automatically deleted in the first reconfigure run af>


^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^/ Go To Line
```

Вернувшись на страницу входа, введите следующее:

- Имя пользователя: root
- Пароль: [пароль указан на /etc/gitlab/initial_root_password]

Введите эти значения в поля и нажмите кнопку «Войти» . Вы войдете в приложение и попадете на целевую страницу, на которой вам будет предложено начать добавлять проекты:

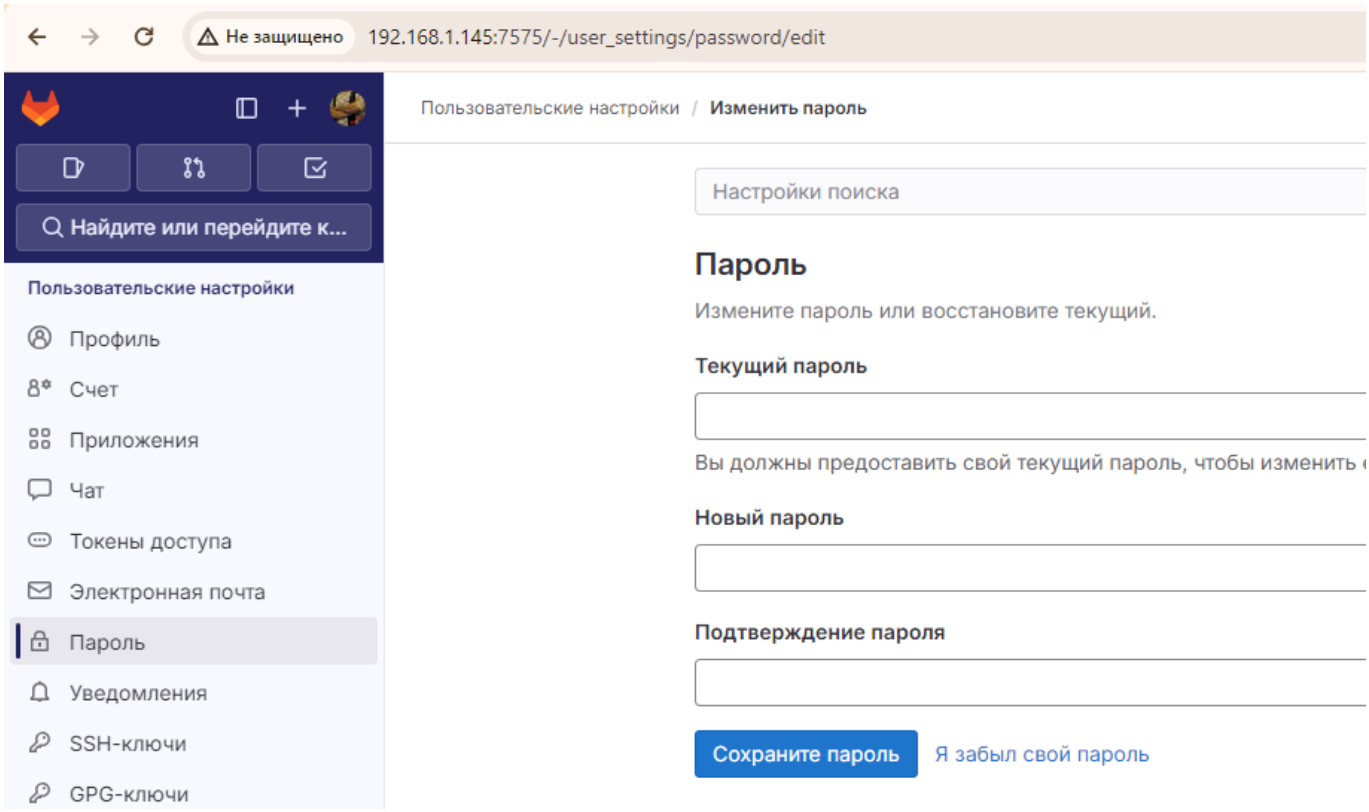


Теперь вы можете точно настроить свой экземпляр GitLab.

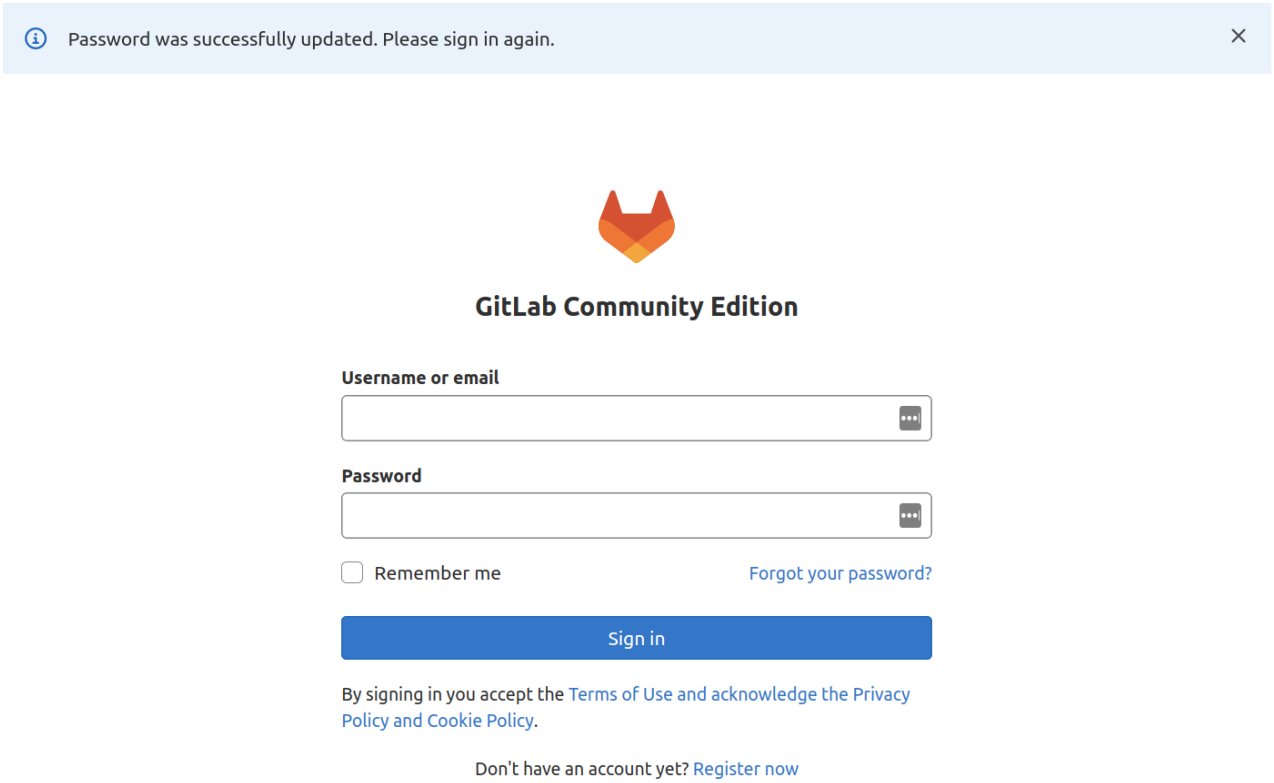
Обновление вашего пароля

Первое, что вам следует сделать после входа в систему, — это сменить пароль. Чтобы внести это изменение, нажмите значок в правом верхнем углу панели навигации и выберите **«Редактировать профиль»:**

Затем вы попадете на страницу настроек пользователя. На левой панели навигации выберите «Пароль» , чтобы изменить пароль, сгенерированный GitLab, на безопасный пароль, а затем нажмите кнопку «Сохранить пароль» , когда закончите вносить обновления:



Вы вернетесь на экран входа в систему с уведомлением о том, что ваш пароль был изменен. Введите новый пароль, чтобы снова войти в свой экземпляр GitLab:



Заключение

Теперь у вас есть работающий экземпляр GitLab, размещенный на вашем собственном сервере. Вы можете начать импортировать или создавать новые проекты и настраивать соответствующий уровень доступа для команды. GitLab регулярно добавляет функции и обновляет свою платформу, поэтому обязательно посещайте домашнюю страницу проекта, чтобы быть в курсе любых улучшений или важных уведомлений.

Ссылки и Дополнения

- оригинальная статья
- Краткое руководство по DNS
- Как настроить брандмауэр с помощью UFW
- Настройка брандмауэра с помощью UFW Ubuntu Server
    - GitLab CE: Community Edition
    - GitLab EE: Enterprise Edition
    - gitlab-ce_16.9.1-ce.0_amd64.deb
    - script.deb.sh

From:
https://wwoss.ru/ - **worldwide open-source software**

Permanent link:
https://wwoss.ru/doku.php?id=software:linux_server:ubuntu_server_install_gitlab

Last update: **2025/12/06 19:58**