

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего
профессионального образования
«Уральский Федеральный Университет имени первого Президента России Б.Н.
Ельцина»
Цифровая кафедра

Отчёт по лабораторной работе №7
Вариант 3. «Установка веб-инструмента Gitlab на Linux Ubuntu»

Студентка	<u>Нюкина Н.А.</u>
Группа	<u>РИ-300013</u>
Дата	<u>03.04.2023</u>

Екатеринбург 2023

Оглавление

ВВЕДЕНИЕ.....	3
ХОД РАБОТЫ	4
Подготовка сервера	4
Установка GitLab	5
Настройка GitLab	8
ЗАКЛЮЧЕНИЕ	13

ВВЕДЕНИЕ

Цель – изучить методы работы с сервисом GitLab.

Задачи:

- подготовить сервер;
- установить GitLab;
- настроить GitLab.

Разрешаем автозапуск сервиса chrony (Рисунок 5):

```
nadya@nadya-VirtualBox:~$ systemctl enable chrony
Synchronizing state of chrony.service with SysV service script with /lib/systemd/
systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable chrony
```

Рисунок 5 – Автозапуск chrony

По умолчанию, в Ubuntu брандмауэр настроен на то, чтобы принимать любые пакеты. Но если у нас он настроен на блокировку, нужно добавить порты 80 и 443 с помощью следующей команд:

```
iptables -I INPUT -p tcp --dport 80 -j ACCEPT
iptables -I INPUT -p tcp --dport 443 -j ACCEPT
```

И чтобы сохранить правила, устанавливаем iptables-persistent (Рисунок 6):

```
nadya@nadya-VirtualBox:~$ sudo apt install iptables-persistent
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
 netfilter-persistent
The following NEW packages will be installed:
 iptables-persistent netfilter-persistent
0 upgraded, 2 newly installed, 0 to remove and 8 not upgraded.
Need to get 13,9 kB of archives.
After this operation, 93,2 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ru.archive.ubuntu.com/ubuntu jammy/universe amd64 netfilter-persistent all 1.0.16 [7 4
40 B]
Get:2 http://ru.archive.ubuntu.com/ubuntu jammy/universe amd64 iptables-persistent all 1.0.16 [6 48
8 B]
Fetched 13,9 kB in 0s (35,5 kB/s)
```

Рисунок 6 – Установка iptables-persistent

И выполняем следующую команду:

```
netfilter-persistent save
```

Результат выполнения команды (Рисунок 7):

```
nadya@nadya-VirtualBox:~$ sudo netfilter-persistent save
run-parts: executing /usr/share/netfilter-persistent/plugins.d/15-ip4tables save
run-parts: executing /usr/share/netfilter-persistent/plugins.d/25-ip6tables save
```

Рисунок 7 – netfilter-persistent

Установка GitLab

Устанавливаем необходимые компоненты (Рисунок 8):

```
nadya@nadya-VirtualBox:~$ sudo apt install curl openssh-server ca-certificates
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20211016ubuntu0.22.04.1).
ca-certificates set to manually installed.
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard monkeysphere ssh-askpass
The following NEW packages will be installed:
  curl ncurses-term openssh-server openssh-sftp-server ssh-import-id
0 upgraded, 5 newly installed, 0 to remove and 8 not upgraded.
Need to get 945 kB of archives.
After this operation, 6 499 kB of additional disk space will be used.
```

Рисунок 8 – Установка необходимых компонентов

Для отправки уведомлений, установим также postfix с помощью следующей команды:

```
sudo apt install postfix
```

При запросе типа конфигурации, выбираем Internet Site (если уведомления должны отправляться наружу) или Local only (уведомления в пределах сервера).

Установим репозиторий GitLab с помощью следующей команды:

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

Результат выполнения команды (Рисунок 9):

```
nadya@nadya-VirtualBox:~$ curl https://packages.gitlab.com/install/repositories/gitlab/g
itlab-ce/script.deb.sh | sudo bash
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 6865 100 6865   0     0  10751      0 --:--:-- --:--:-- --:--:-- 10743
Detected operating system as Ubuntu/jammy.
Checking for curl...
Detected curl...
Checking for gpg...
Detected gpg...
Running apt-get update... done.
Installing apt-transport-https... done.
Installing /etc/apt/sources.list.d/gitlab_gitlab-ce.list...done.
Importing packagecloud gpg key... done.
Running apt-get update... done.

The repository is setup! You can now install packages.
```

Рисунок 9 – Скачивание репозитория GitLab

Устанавливаем Gitlab с помощью следующей команды:

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

Результат выполнения команды (Рисунок 10):



Рисунок 10 – Установка GitLab

Для запуска и корректной работы портала мы должны задать `external_url`. Для этого открываем файл:

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

Изменяем параметр `external_url` (Рисунок 11):

```
external_url 'http://gitlab.site.ru'
```

Рисунок 11 – Редактирование файла /etc/gitlab/gitlab.rb

Выполняем конфигурирование с помощью следующей команды:

```
gitlab-ctl reconfigure
```

Смотрим пароль, который был назначен пользователю после установки в файле `/etc/gitlab/initial_root_password` с помощью следующей команды:

```
cat /etc/gitlab/initial_root_password | grep Password:
```

Открываем браузер и вводим наш адрес, который мы указали в настройках в опции `external_url` — `http://gitlab.site.ru`. Мы должны увидеть страницу авторизации, на которой нас попросят пароль для администратора. Входим в качестве пользователя `root` с паролем, который посмотрели в файле (Рисунок 12):



GitLab Community Edition

Username or email

Password

Remember me [Forgot your password?](#)

[Sign in](#)

[Don't have an account yet? Register now](#)

Рисунок 12 – Вход в GitLab

Настройка GitLab

По умолчанию, портал устанавливается с интерфейсом на английском. Для смены языка, кликаем по иконке в правом верхнем углу и выбираем Preferences (Рисунок 13):

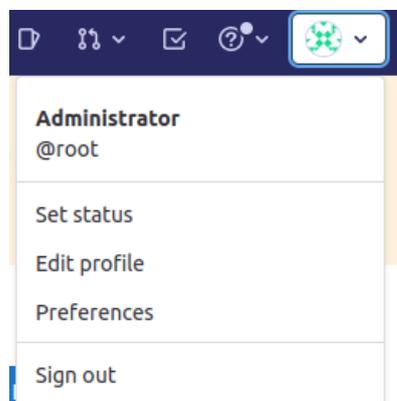


Рисунок 13 – Открытие настроек

В подразделе Localization выбираем нужный нам язык и первый день недели (Рисунок 14):

Localization
Customize language and region related settings. [Learn more.](#)

Language

Russian - русский (25% translated) ▾

This feature is experimental and translations are not yet complete.
[Help translate GitLab into your language](#) ↗

First day of the week

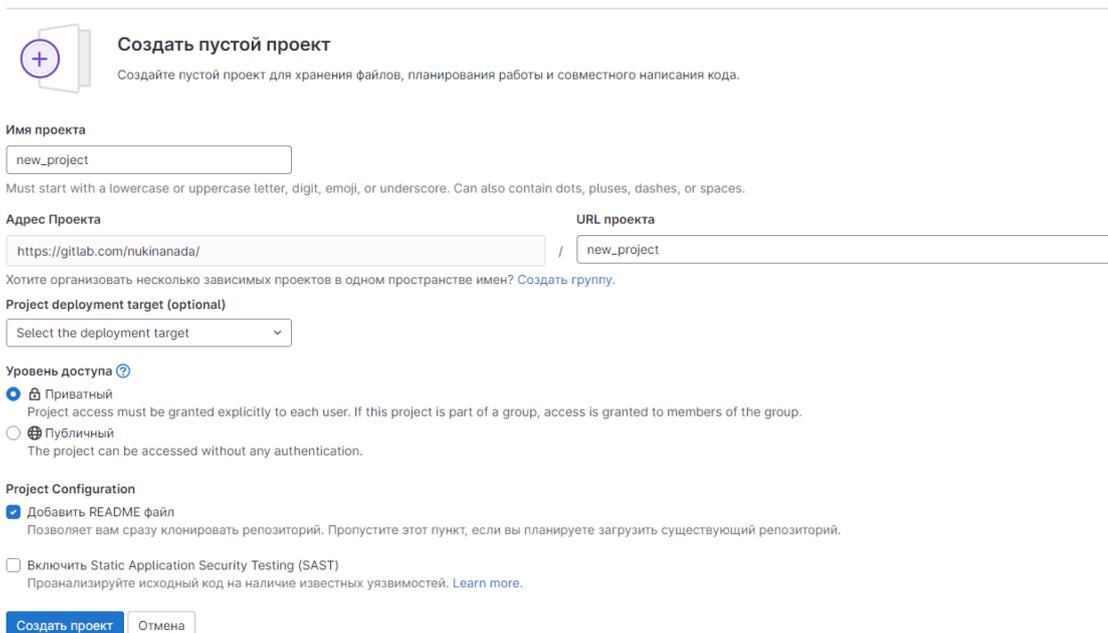
Monday ▾

Рисунок 14 – Настройка языка

Сохраняем настройки и перезапускаем страницу для применения нового языка.

Попробуем создать проект и подключиться к нему из Linux. Также для теста мы создадим файл и закинем его в наш репозиторий.

В веб-интерфейсе GitLab создаем новый проект (Рисунок 15):



Создать пустой проект
Создайте пустой проект для хранения файлов, планирования работы и совместного написания кода.

Имя проекта

Must start with a lowercase or uppercase letter, digit, emoji, or underscore. Can also contain dots, pluses, dashes, or spaces.

Адрес Проекта / URL проекта

Хотите организовать несколько зависимых проектов в одном пространстве имен? [Создать группу](#).

Project deployment target (optional)

Уровень доступа

Приватный
Project access must be granted explicitly to each user. If this project is part of a group, access is granted to members of the group.

Публичный
The project can be accessed without any authentication.

Project Configuration

Добавить README файл
Позволяет вам сразу клонировать репозиторий. Пропустите этот пункт, если вы планируете загрузить существующий репозиторий.

Включить Static Application Security Testing (SAST)
Проанализируйте исходный код на наличие известных уязвимостей. [Learn more](#).

Рисунок 15 – Создание и настройка нового проекта

Для примера попробуем подключиться с компьютера Linux к нашему репозиторию и закинуть на него тестовый файл.

Для начала установим git на компьютер с Linux (Рисунок 16):

```
nadya@nadya-VirtualBox:~$ sudo apt install git
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  git-man liberror-perl
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-email git-gui gitk gitweb
  git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  git git-man liberror-perl
0 upgraded, 3 newly installed, 0 to remove and 13 not upgraded.
Need to get 4 121 kB of archives.
After this operation, 20,9 MB of additional disk space will be used.
```

Рисунок 16 – Установка git

Создаем папку для тестового проекта (Рисунок 17):

```
nadya@nadya-VirtualBox:~$ mkdir -p projects/test
nadya@nadya-VirtualBox:~$ ls
Desktop  Downloads  Pictures  Public  Templates
Documents Music      projects  snap    Videos
```

Рисунок 17 – Создание папки для тестового проекта

Переходим в нее с помощью следующей команды:

```
cd /projects/test
```

Создаем репозиторий (Рисунок 18):

```
nadya@nadya-VirtualBox:~/projects/test$ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/nadya/projects/test/.git/
```

Рисунок 18 – Создание репозитория

Создаем файл (Рисунок 19):

```
nadya@nadya-VirtualBox:~/projects/test$ touch testfile.txt
nadya@nadya-VirtualBox:~/projects/test$ ls
testfile.txt
```

Рисунок 19 – Создание файла

Добавляем все файлы в индекс (Рисунок 20):

```
nadya@nadya-VirtualBox:~/projects/test$ git add .
nadya@nadya-VirtualBox:~/projects/test$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   testfile.txt
```

Рисунок 20 – Добавление файла в индекс

Делаем коммит с помощью следующей команды:

```
git commit -m "Commit 1" -a
```

Подключаемся к созданному репозиторию с помощью следующей команды:

```
git remote add origin http://gitlab.com/nukinanada/new_project
```

Заливаем в него закоммиченный файл с помощью следующей команды:

```
git push origin master
```

Переходим на веб-страницу нашего проекта — мы должны увидеть наш файл (Рисунок 21):

Наименование	Последний коммит	Последнее обновление
• README.md	Initial commit	5 минут назад
• testfile.txt	Commit 1	Только что

Рисунок 21 – Созданный файл

Настройка SSL

В данном примере мы сконфигурируем наш сервер для возможности работы по https и получения сертификата от Let's Encrypt. Все настройки выполняются в конфигурационном файле. Открываем его (Рисунок 22):

```
nadya@nadya-VirtualBox:~$ sudo nano /etc/gitlab/gitlab.rb
```

Рисунок 22 – Открытие конфигурационного файла

Вносим изменения в файл (Рисунок 23):

```
external_url 'https://gitlab.site.ru'
```

Рисунок 23 – Изменения конфигурационного файла

Также настраиваем получение сертификата от Let's Encrypt (Рисунок 24):

```
letsencrypt['enable'] = true
```

Рисунок 24 – Настройка получения сертификата от Let's Encrypt

Также задаем следующие настройка (Рисунок 25):

```
nginx['redirect_http_to_https'] = true

letsencrypt['auto_renew'] = true
letsencrypt['auto_renew_hour'] = "22"
letsencrypt['auto_renew_minute'] = "50"
letsencrypt['auto_renew_day_of_month'] = "* /7"

nginx['ssl_certificate'] = "/etc/gitlab/ssl/#{node['fqdn']}.crt"
nginx['ssl_certificate_key'] = "/etc/gitlab/ssl/#{node['fqdn']}.key"
```

Рисунок 25 – Настройка конфигурации

Создаем вышеуказанный каталог (Рисунок 26):

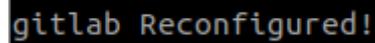
```
nadya@nadya-VirtualBox:~$ mkdir -p /etc/gitlab/ssl/
```

Рисунок 26 – Создание каталога /etc/gitlab/ssl

Применяем новую конфигурацию с помощью следующей команды:

```
sudo gitlab-ctl reconfigure
```

Конфигурация применена успешно (Рисунок 27):



```
gitlab Reconfigured!
```

Рисунок 27 – Результаты конфигурации

Если мы забыли пароль для пользователя root, можно его сбросить через командную строку.

Подключаемся к консоли управления gitlab с помощью команды:

```
gitlab-rails console -e production
```

Создаем переменную, которая будет вести на ссылку с учетной записью root (идентификатор 1):

```
user = User.where(id: 1).first
```

Задаем пароль для пользователя root дважды:

```
user.password = 'password123'  
user.password_confirmation = 'password123'
```

Сохраняем изменения для пользователя:

```
user.save!
```

Готово.

ЗАКЛЮЧЕНИЕ

В ходе лабораторной работы мною был изучен процесс работы с сервисом GitLab.