

Лабораторная работа №2

Первоначальная настройка git

Мухин Тимофей

02.03.2024

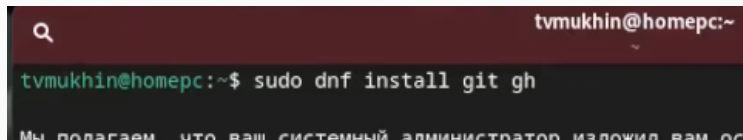
Российский университет дружбы народов, Москва, Россия

Цель работы

Изучить идеологию и применение средств контроля версий. Освоить умения по работе с git

Выполнение работы

Установим git и gh из репозитория

A terminal window with a dark red title bar. The title bar contains a magnifying glass icon on the left and the text 'tvmukhin@homepc:~' on the right. The terminal content shows a green prompt 'tvmukhin@homepc:~\$' followed by the command 'sudo dnf install git gh' in white text. Below the command, the beginning of a system message is visible: 'Мы полагаем, что ваш системный администратор изложил вам ос'.

```
tvmukhin@homepc:~$ sudo dnf install git gh
```

Мы полагаем, что ваш системный администратор изложил вам ос

Базовая настройка git. Зададим имя и email владельца репозитория

```
tvmukhin@homepc:~$ git config --global user.name "Timofey"  
tvmukhin@homepc:~$ git config --global user.email "video2244@gmail.com"
```

Настроим верификацию и подписание комитов, зададим имя начальной ветки, параметры autocrlf и safecrlf

```
tvmukhin@homepc:~$ git config --global core.quotePath false
tvmukhin@homepc:~$ git config --global init.defaultBranch master
tvmukhin@homepc:~$ git config --global core.autocrlf input
tvmukhin@homepc:~$ git config --global core.safecrlf warn
tvmukhin@homepc:~$
```

Создаем ключи ssh rsa и ed25519

```
tvmukhin@homepc:~$ ssh-keygen -t ed25519  
Generating public/private ed25519 key pair.  
Enter file in which to save the key (/home/tvmukhin/.ssh/id_ed25519):
```


Создаем ключи gpg и добавляем ключ в github

```
tvmukhin@homepc:~$ gpg --full-generate-key
gpg (GnuPG) 2.4.3; Copyright (C) 2023 g10 Code GmbH
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
```

```
gpg: создан каталог '/home/tvmukhin/.gnupg'
```

Выберите тип ключа:

- (1) RSA and RSA
- (2) DSA and Elgamal
- (3) DSA (sign only)
- (4) RSA (sign only)
- (9) ECC (sign and encrypt) *default*
- (10) ECC (только для подписи)
- (14) Existing key from card

Ваш выбор? 1

длина ключей RSA может быть от 1024 до 4096.

Какой размер ключа Вам необходим? (3072) 4096

Запрошенный размер ключа - 4096 бит

Выберите срок действия ключа.

0 = не ограничен

<n> = срок действия ключа - n дней

<n>w = срок действия ключа - n недель

<n>m = срок действия ключа - n месяцев

<n>y = срок действия ключа - n лет

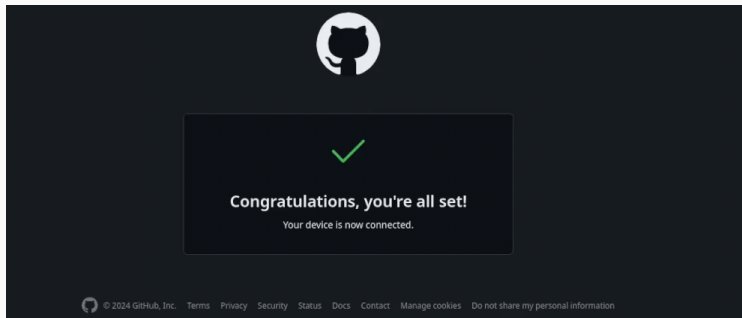
Используя введённый email, укажем Git применять его при подписи коммитов

```
bash: syntax error near unexpected token `newline'
tvmukhin@homepc:~$ git config --global user.signingkey
tvmukhin@homepc:~$ git config --global user.signingkey key1
tvmukhin@homepc:~$ git config --global commit.gpgsign true
tvmukhin@homepc:~$
```

Настройка gh. Авторизуемся.

```
tvmukhin@homepc:~$ gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations on this host? [Use arrows to move, type to filter]
> HTTPS
SSH
```

Настройка gh. Авторизуемся.



Создание репозитория курса на основе шаблона. Создаем директорию.

```
tvmukhin@homepc:~$ mkdir -p ~/work/study/2023-2024/"Операционные_системы"/os-intro  
tvmukhin@homepc:~$
```

Создание репозитория курса на основе шаблона. Создаем директорию.

```
tvmukhin@homepc:~$ cd ~/work/study/2023-2024/Операционные_системы/os-intro/  
tvmukhin@homepc:~/work/study/2023-2024/Операционные_системы/os-intro$ gh repo create study_2023-2024_os-intro --template=yamadharma/course-directory-student-template --public  
✓ Created repository Seychik1/study_2023-2024_os-intro on GitHub  
https://github.com/Seychik1/study_2023-2024_os-intro  
tvmukhin@homepc:~/work/study/2023-2024/Операционные_системы/os-intro$ git clone
```

Клонируем репозиторий на компьютер

```
и репозиторий существует.  
tvmukhin@homepc:~/work/study/2023-2024/Операционные_системы/os-intro$ git clone --recurse https://github.com/Seychik1/study_2023-2024_os-intro.git os-intro  
Клонирование в «os-intro»...
```

Настройка каталога курса. Удаляем лишние файлы, создаем необходимые каталоги

```
os-intro
tvmukhin@homepc:~/work/study/2023-2024/Операционные_системы/os-intro$ cd os-intro/
tvmukhin@homepc:~/work/study/2023-2024/Операционные_системы/os-intro/os-intro$ ls
CHANGELOG.md  COURSE  Makefile  README.en.md  README.md
config        LICENSE  package.json  README.git-flow.md  template
tvmukhin@homepc:~/work/study/2023-2024/Операционные_системы/os-intro/os-intro$ rm package.json
tvmukhin@homepc:~/work/study/2023-2024/Операционные_системы/os-intro/os-intro$ echo os-intro > CO
E
tvmukhin@homepc:~/work/study/2023-2024/Операционные_системы/os-intro/os-intro$ make
Usage:
  make <target>

Targets:
  list           List of courses
  prepare       Generate directories structure
  submodule     Update submodules
```


Делаем коммит и отправляем данные на сервер

```
tvmukhin@homepc: ~/work/study/2023-2024/Операционные_системы/os-intro$ git commit -am 'feat(main): make course structure'
```

Делаем коммит и отправляем данные на сервер

```
tvmukhin@homepc:~/work/study/2023-2024/Операционные_системы/os-intro$ git push
```

Вывод

В ходе выполнения работы было изучено применение и работа системы контроля версий git