

# **Отчет о выполнении лабораторной работы 2**

**Операционные системы**

Пестова Ева Константиновна

# Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	18

## Список иллюстраций

3.1	Установка git . . . . .	7
3.2	Установка git . . . . .	7
3.3	Установка gh . . . . .	8
3.4	Базовая настройка git . . . . .	8
3.5	Создание ключа по алгоритму rsa . . . . .	9
3.6	Создание ключа по алгоритму ed25519 . . . . .	9
3.7	Генерация ключа . . . . .	9
3.8	Заполнение данных . . . . .	10
3.9	Создание защитного пароля . . . . .	10
3.10	Аккаунт на github . . . . .	11
3.11	Список ключей . . . . .	11
3.12	Отпечаток ключа . . . . .	12
3.13	Добавление ключа . . . . .	12
3.14	Ключ добавился . . . . .	12
3.15	Настройка подписей . . . . .	13
3.16	Настройка gh . . . . .	13
3.17	Активация по коду . . . . .	14
3.18	Результат . . . . .	15
3.19	Создание каталога . . . . .	15
3.20	Каталог создан . . . . .	15
3.21	Создание репозитория . . . . .	16
3.22	Создание репозитория . . . . .	16
3.23	Настройка каталога круса . . . . .	16
3.24	Отправление файлов на github . . . . .	17

## Список таблиц

# 1 Цель работы

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

## 2 Задание

1. Создать базовую конфигурацию для работы с git.
2. Создать ключ SSH.
3. Создать ключ PGP.
4. Настроить подписи git.
5. Зарегистрироваться на Github.
6. Создать локальный каталог для выполнения заданий по предмету

### 3 Выполнение лабораторной работы

1. Создание базовой конфигурации для работы с git.

Первым делом я устанавливаю git (рис. 3.1), (рис. 3.2).

```
(ekpestova@kali)-[~]  
$ sudo apt-get update  
Get:1 http://kali.download/kali kali-rolling InRelease [41.5 kB]  
Get:2 http://kali.download/kali kali-rolling/main amd64 Packages [19.8 MB]  
Get:3 http://kali.download/kali kali-rolling/main amd64 Contents (deb) [47.2 MB]  
Fetched 67.1 MB in 17s (4064 kB/s)  
Reading package lists... Done  
  
(ekpestova@kali)-[~]  
$ sudo apt install git
```

Рис. 3.1: Установка git

```
(ekpestova@kali)-[~]  
$ sudo apt-get update  
Hit:1 http://kali.download/kali kali-rolling InRelease  
Reading package lists... Done  
  
(ekpestova@kali)-[~]  
$ sudo apt install git-all
```

Рис. 3.2: Установка git

Далее я устанавливаю gh (рис. 3.3).

```
(ekpestova@kali)-[~]
└─$ sudo apt install gh
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  gh
0 upgraded, 1 newly installed, 0 to remove and 2 not up
graded.
Need to get 7982 kB of archives.
After this operation, 45.3 MB of additional disk space w
ill be used.
Get:1 http://kali.download/kali kali-rolling/main amd64
gh amd64 2.44.1-1 [7982 kB]
Fetched 7982 kB in 2s (4962 kB/s)
```

Рис. 3.3: Установка gh

Затем, провожу базовую настройку git: задаю имя и email владельца репозитория, настраиваю utf-8 в выводе сообщений git, задаю имя начальной ветки, настраиваю параметры autocrlf и safecrlf (рис. 3.4).

```
(ekpestova@kali)-[~]
└─$ git config --global user.name "Eva Pestova"

(ekpestova@kali)-[~]
└─$ git config --global user.email "1132236053@pfur.ru"

(ekpestova@kali)-[~]
└─$ git config --global core.quotepath false

(ekpestova@kali)-[~]
└─$ git config --global init.defaultBranch master

(ekpestova@kali)-[~]
└─$ git config --global core.autocrlf input

(ekpestova@kali)-[~]
└─$ git config --global core.safecrlf warn
```

Рис. 3.4: Базовая настройка git

## 2. Создание ключей SSH

Я создаю ключи SSH по алгоритмам rsa и ed25519 (рис. 3.5), (рис. 3.6).



```

(ekpestova@kali)-[~]
$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ekpestova/.ssh/id_rsa):
Created directory '/home/ekpestova/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again: █

```

Рис. 3.5: Создание ключа по алгоритму rsa

```

(ekpestova@kali)-[~]
$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/ekpestova/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase): █

```

Рис. 3.6: Создание ключа по алгоритму ed25519

### 3. Создание ключей pgr

Следующим шагом я генерирую pgr ключ, выбираю нужные опции и ввожу необходимые данные: тип RSA and RSA, размер 4096, срок действия не истекает никогда, ввожу имя и свою почту (рис. 3.7), (рис. 3.8).

```

(ekpestova@kali)-[~]
$ gpg --full-generate-key
gpg (GnuPG) 2.2.40; Copyright (C) 2022 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: keybox '/home/ekpestova/.gnupg/pubring.kbx' created
Please select what kind of key you want:
  (1) RSA and RSA (default)
  (2) DSA and Elgamal
  (3) DSA (sign only)
  (4) RSA (sign only)
  (14) Existing key from card
Your selection? RSA
Invalid selection.
Your selection? 1
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (3072) 4096 █

```

Рис. 3.7: Генерация ключа

```
Key is valid for? (0)
Key does not expire at all
Is this correct? (y/N) y

GnuPG needs to construct a user ID to identify your key.

Real name: ekpestova
Email address: 1132236053@pfur.ru
Comment:
You selected this USER-ID:
    "ekpestova <1132236053@pfur.ru>"
```

Рис. 3.8: Заполнение данных

Далее я придумываю пароль для защиты нового ключа (рис. 3.9).

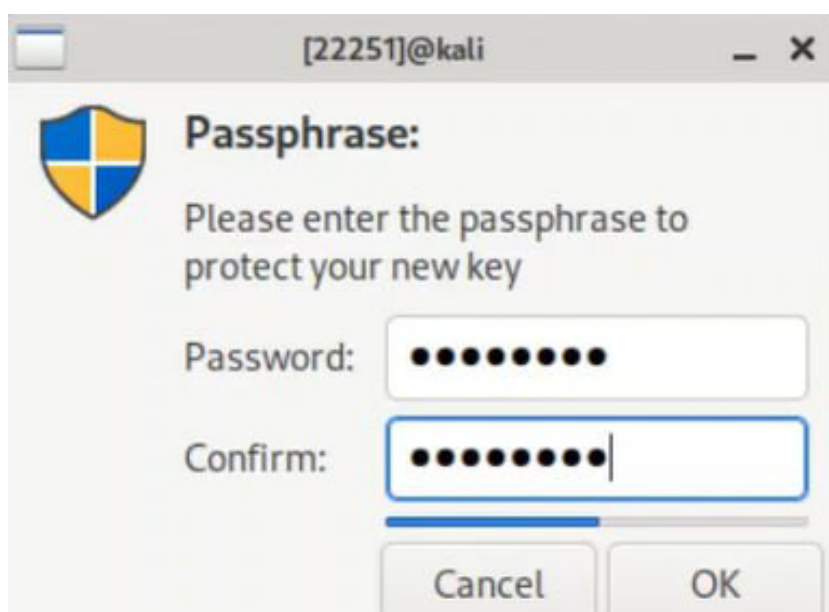


Рис. 3.9: Создание защитного пароля

#### 4. Регистрация на github

Так как у меня уже есть аккаунт на github, я просто вхожу в него (рис. 3.10).

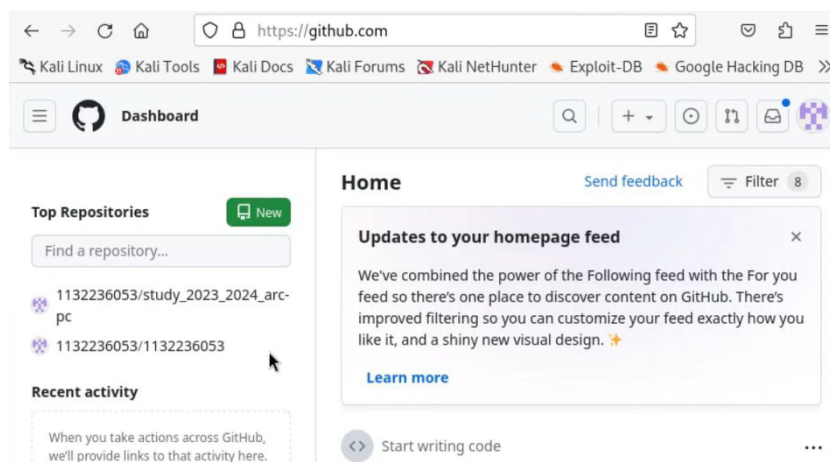


Рис. 3.10: Аккаунт на github

## 5. Добавление pgp ключа в github

Вывожу список ключей (рис. 3.11).

```

File Actions Edit View Help
sub  rsa4096 2024-02-25 [E]

(ekpestova@kali)-[~]
$ gpg --list-secret-keys --keyid-format LONG
gpg: checking the trustdb
gpg: marginals needed: 3  completes needed: 1  trust mod
el: pgp
gpg: depth: 0  valid: 1  signed: 0  trust: 0-, 0q, 0
n, 0m, 0f, 1u
/home/ekpestova/.gnupg/pubring.kbx

sec  rsa4096/4784841AE8B9241B 2024-02-25 [SC]
     EA53FD06983F2C89F7FB101C4784841AE8B9241B
uid                               [ultimate] ekpestova <1132236053@pfu
r.ru>
ssb  rsa4096/FC10D47D3A0EF9D1 2024-02-25 [E]

```

Рис. 3.11: Список ключей

Копирую отпечаток приватного ключа (рис. 3.12).

```
(ekpestova@kali)-[~]
$ gpg --armor --export 4784841AE8B9241B | xclip -sel c
ip
```

Рис. 3.12: Отпечаток ключа

Добавляю новый gpg ключ на github (рис. 3.13).

## Add new GPG key

Title

Key

```
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQINBGXbRxIBeADiJtC/COFSXE0qH
/8+Zd48jZwNFCh7VppMwtfx50vZAcSPJ/v+
rQf/IjqR4ZAqWFgKjory
```

Рис. 3.13: Добавление ключа

Мы видим, что добавился наш ключ: появились subkeys и ID ключа (рис. 3.14).

## GPG keys

New GPG key

This is a list of GPG keys associated with your account. Remove any keys that you do not recognize.



**Email address:** 1132236053@pfur.ru  
**Key ID:** 4784841AE8B9241B  
**Subkeys:** FC10D47D3A0EF9D1  
 Added on Feb 25, 2024

Delete

Learn how to [generate a GPG key and add it to your account](#).

Рис. 3.14: Ключ добавился

## 6. Настройка подписей git

Далее, я настраиваю автоматические подписи коммитов git (рис. 3.15).

```
(ekpestova@kali)-[~]  
$ git config --global user.signingkey 4784841AE8B9241B  
  
(ekpestova@kali)-[~]  
$ git config --global commit.gpgsign true  
  
(ekpestova@kali)-[~]  
$ git config --global gpg.program $(which gpg2)
```

Рис. 3.15: Настройка подписей

## 7. Настройка gh

Я авторизуюсь и копирую код (рис. 3.16).

```
(ekpestova@kali)-[~]  
$ gh auth login  
? What account do you want to log into? GitHub.com  
? What is your preferred protocol for Git operations on this host? HTTPS  
? Authenticate Git with your GitHub credentials? Yes  
? How would you like to authenticate GitHub CLI? Login with a web browser  
  
! First copy your one-time code: 8149-F73B  
Press Enter to open github.com in your browser ...
```

Рис. 3.16: Настройка gh

Ввожу код на странице github, которая открылась автоматически (рис. 3.17).

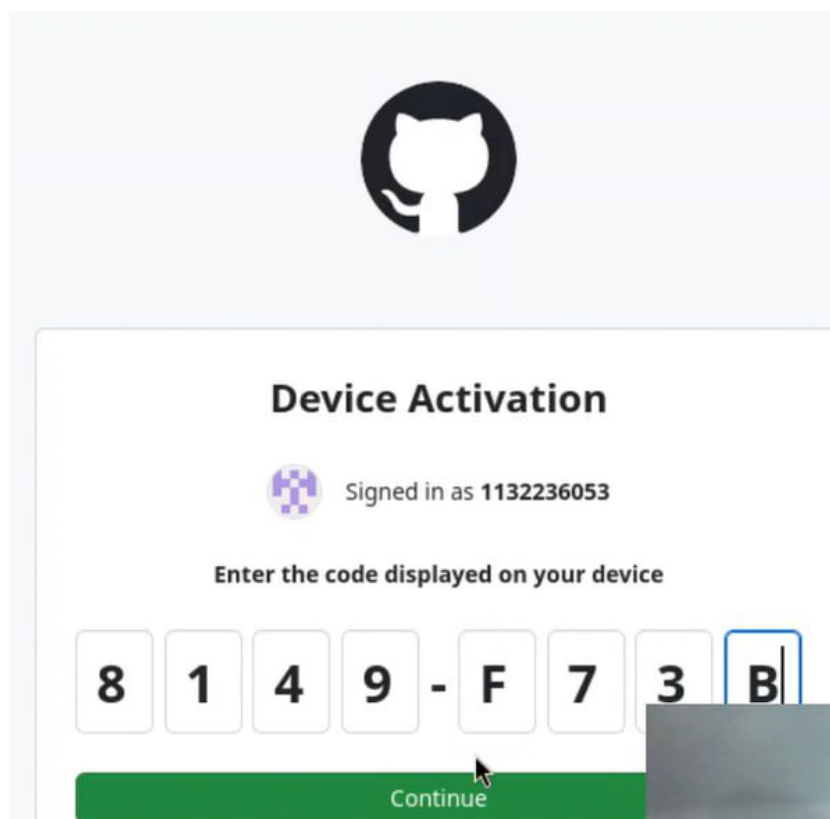


Рис. 3.17: Активация по коду

Как мы видим, нам удалось авторизоваться (рис. 3.18).

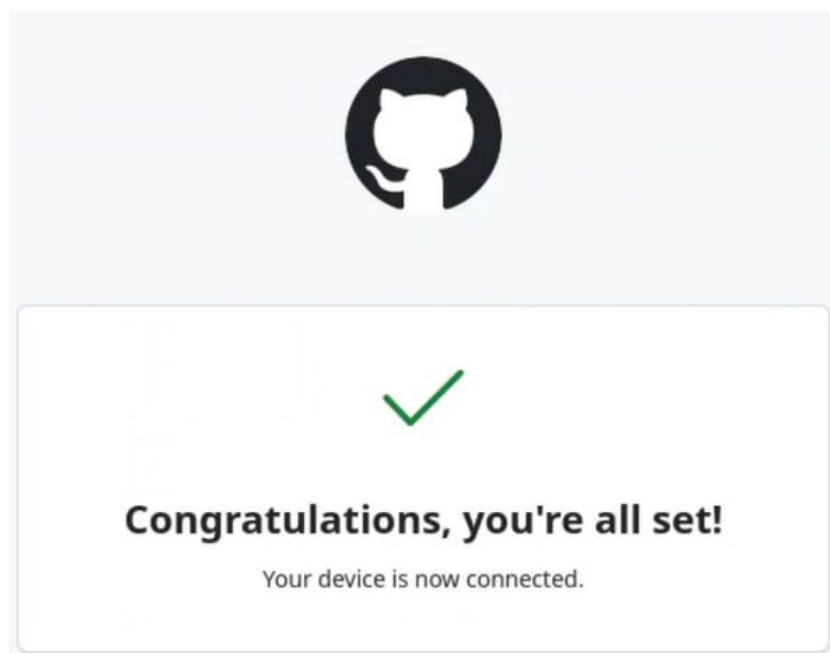


Рис. 3.18: Результат

## 8. Создание репозитория круса на основе шаблона

Сначала я создаю каталог в домашней папке (рис. 3.19), (рис. 3.20).

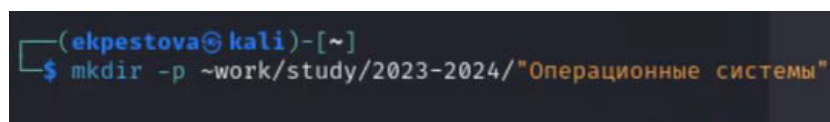


Рис. 3.19: Создание каталога

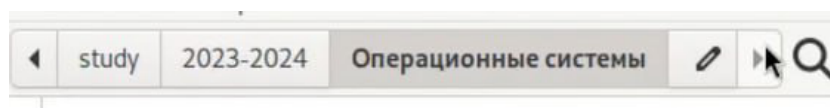


Рис. 3.20: Каталог создан

Перехожу в него и создаю такой же репозиторий на github (рис. 3.21), (рис. 3.22).

```

(ekpestova@kali)-[~]
└─$ cd ~work/study/2023-2024/"Операционные системы"

(ekpestova@kali)-[~/~work/study/2023-2024/Операционные системы]
└─$ gh repo create study_2023-2024_os-intro --template=yamadharma/course-directory-student-template --public
✓ Created repository 1132236053/study_2023-2024_os-intro on GitHub
https://github.com/1132236053/study_2023-2024_os-intro

```

Рис. 3.21: Создание репозитория

```

(ekpestova@kali)-[~/~work/study/2023-2024/Операционные системы]
└─$ git clone --recursive https://github.com/1132236053/study_2023-2024_os-intro os-intro

```

Рис. 3.22: Создание репозитория

Далее, я настраиваю каталог курса: удаляю лишние файлы и создаю необходимые каталоги (рис. 3.23).

```

(ekpestova@kali)-[~]
└─$ cd ~work/study/2023-2024/"Операционные системы"/os-intro

(ekpestova@kali)-[~/~/study/2023-2024/Операционные системы/os-intro]
└─$ rm package.json

(ekpestova@kali)-[~/~/study/2023-2024/Операционные системы/os-intro]
└─$ echo os-intro > COURSE

(ekpestova@kali)-[~/~/study/2023-2024/Операционные системы/os-intro]
└─$ make

```

Рис. 3.23: Настройка каталога курса

В конце лабораторной работы я отправляю файлы на сервер (рис. 3.24).



```
(ekpestova@kali)-[~/../study/2023-2024/Операционные системы/os-intro]
└─$ git add .

(ekpestova@kali)-[~/../study/2023-2024/Операционные системы/os-intro]
└─$ git commit -am 'feat(main): make course structure'
[master a5dc1fc] feat(main): make course structure
2 files changed, 1 insertion(+), 14 deletions(-)
delete mode 100644 package.json

(ekpestova@kali)-[~/../study/2023-2024/Операционные системы/os-intro]
└─$ git push
```

Рис. 3.24: Отправление файлов на github

## 4 Выводы

Мне удалось изучить идеологию и применение средств контроля версий, а также Освоить умения по работе с git.