

лабораторная работа №2

Операционные системы

Павлова Татьяна Юрьевна

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
3.1	Установка программного обеспечения	7
3.2	Базовая настройка git	7
3.3	Создание ключа GPG	9
3.4	Регистрация на GitHub	10
3.5	Добавление ключа GPG в GitHub	11
3.6	Настройки подписи Git	12
3.7	Настройка gh	12
3.8	Создание репозитория курса на основе шаблона	12
4	Выводы	14
	Список литературы	15

Список иллюстраций

3.1	Установка	7
3.2	Заданию имя и email	7
3.3	настройка	7
3.4	Задаю имя начальной ветке	8
3.5	Задаю параметры	8
3.6	создание ключа	8
3.7	Создание ключа	9
3.8	Генерация ключа	10
3.9	Мой аккаунт	10
3.10	Копирование ключа	11
3.11	Копирование ключа	11
3.12	Натсройки github	11
3.13	Добавление нового ключа	11
3.14	Настраивание подписей	12
3.15	Авторизация	12
3.16	Сообщение о завершении авторизации	12
3.17	Создание директории и репозитория	13
3.18	Клонирование	13
3.19	Удаление и создание	13

Список таблиц

1 Цель работы

цель данной работы - изучение идеологии и применения средств контроля версий, освоение умения по работе с git.

2 Задание

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

3 Выполнение лабораторной работы

3.1 Установка программного обеспечения

Устанавливаю необходимое программное обеспечение через терминал (рис. 3.1).

```
[tanya@tatyana-pavlova ~]$ sudo dnf -y install git
[sudo] пароль для tanya:
Обновление и загрузка репозитория:
Репозитории загружены.
Пакет "git-2.48.1-1.fc41.x86_64" уже установлен.

Нечего делать.
[tanya@tatyana-pavlova ~]$ sudo dnf -y install gh
Обновление и загрузка репозитория:
Репозитории загружены.
Пакет
Установка:
Арх.
Версия
Репозиторий
Размер
gh
x86_64
2.65.0-1.fc41
updates
42.6 MiB
Сводка транзакции:
Установка: 1 пакета
Общий размер входящих пакетов составляет 10 MiB. Необходимо загрузить 10 MiB.
```

Рис. 3.1: Установка

3.2 Базовая настройка git

Задаю имя и email (рис. 3.2).

```
[tanya@tatyana-pavlova ~]$ git config --global user.name "Tatyana Pavlova"
[tanya@tatyana-pavlova ~]$ git config --global user.email "1132246745@pfur.ru"
[tanya@tatyana-pavlova ~]$
```

Рис. 3.2: Заданию имя и email

Настраиваю utf-8 (рис. 3.3).

```
[tanya@tatyana-pavlova ~]$ git config --global core.quotePath false
[tanya@tatyana-pavlova ~]$
```

Рис. 3.3: настройка

Начальной ветке задаю имя master (рис. 3.4).

```
[tanya@tatyana-pavlova ~]$ git config --global init.defaultBranch master
[tanya@tatyana-pavlova ~]$
```

Рис. 3.4: Задаю имя начальной ветке

Задаю параметры autocrlf и safecrlf для корректного отображения конца строки (рис. 3.5).

```
[tanya@tatyana-pavlova ~]$ git config --global core.autocrlf input
[tanya@tatyana-pavlova ~]$ git config --global core.safecrlf warn
[tanya@tatyana-pavlova ~]$
```

Рис. 3.5: Задаю параметры

Создаю ключ ssh размером 4096 бит по алгоритму rsa (рис. 3.6).

```
[tanya@tatyana-pavlova ~]$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/tanya/.ssh/id_rsa):
Created directory '/home/tanya/.ssh'.
Enter passphrase for "/home/tanya/.ssh/id_rsa" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/tanya/.ssh/id_rsa
Your public key has been saved in /home/tanya/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:RQTOXmoBYhOoGeKdsRjfmAeyLrrNGS+NHnsQ10Tf/JM tanya@tatyana-pavlova
The key's randomart image is:
+---[RSA 4096]-----+
|  .oo .oo          |
|oo.+ o.= +         |
|o+B Oo = =         |
|o+.B.o.. = . .     |
|. o. S E           |
|... . . .          |
|o o+               |
|.oo*o              |
|.o*o.              |
+---[SHA256]-----+
[tanya@tatyana-pavlova ~]$
```

Рис. 3.6: создание ключа

Создаю ключ ssh по алгоритму ed25519 (рис. 3.7).


```
[tanya@tatyapavlova ~]$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/tanya/.ssh/id_ed25519):
Enter passphrase for "/home/tanya/.ssh/id_ed25519" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/tanya/.ssh/id_ed25519
Your public key has been saved in /home/tanya/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:sSE7jqER6CXHQaDXU2SM5xpk4164chJi8n6fDZr3SvE tanya@tatyapavlova
The key's randomart image is:
+--[ED25519 256]--+
| .oo ++          |
| ....*oo         |
| o.oBo=. o       |
| +o=.+.oo +      |
| o+.o.=+ S       |
|  +o=+ +         |
| ..+. + E        |
|  .+.+          |
|  .oo+oo         |
+-----[SHA256]-----+
```

Рис. 3.7: Создание ключа

3.3 Создание ключа GPG

Генерирую ключ gpg, затем выбираю тип ключа rsa and rsa, задаю максимальную длину ключа:4096, оставляю неограниченный срок действия (рис. 3.8).

```

[tanya@tatyana-pavlova ~]$ gpg --full-generate-key
gpg (GnuPG) 2.4.5; Copyright (C) 2024 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/tanya/.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
Срок действия ключа? (0) 0
Срок действия ключа не ограничен
Все верно? (y/N) y

```

Рис. 3.8: Генерация ключа

3.4 Регистрация на GitHub

У меня уже создан аккаунт, поэтому я буду использовать его (рис. 3.9).

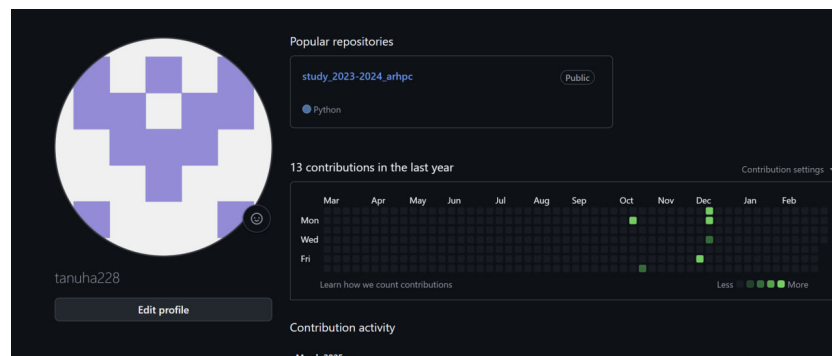


Рис. 3.9: Мой аккаунт

3.5 Добавление ключа GPG в GitHub

Вывожу список созданных ключей с помощью команды и копирую нужную мне часть в буфер обмена (рис. 3.10).

```
[tanya@tatyana-pavlova ~]$ gpg --list-secret-keys --keyid-format LONG
gpg: проверка таблицы доверия
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: глубина: 0 достоверных: 1 подписанных: 0 доверие: 0-, 0q, 0n, 0m, 0f, 1u
[keyboard]
-----
sec   rsa4096/FC4A2C7B78CA6ECE 2025-03-05 [SC]
      5FA8344935BE077484DA7D21FC4A2C7B78CA6ECE
uid           [ абсолютно ] PavlovaTatyana <1132246745@pfur.ru>
ssb   rsa4096/C7E9E3BEEBA18E5B 2025-03-05 [E]
```

Рис. 3.10: Копирование ключа

Ввожу в терминал команду, с помощью которой копирую сам ключ (рис. 3.11).

```
[tanya@tatyana-pavlova ~]$ gpg --armor --export FC4A2C7B78CA6ECE | xclip -sel clip
```

Рис. 3.11: Копирование ключа

Ищу в настройках github добавление gpg ключа (рис. 3.12).



Рис. 3.12: Настройки github

Вставляю в поле ключ из буфера обмена (рис. 3.13).

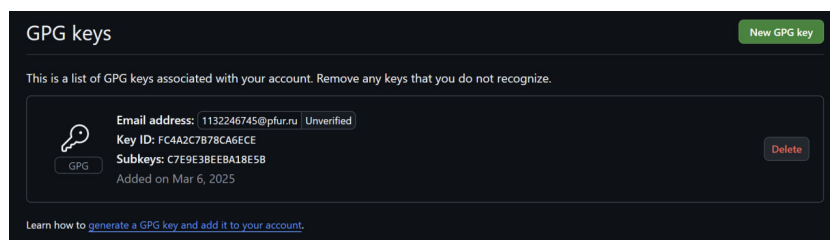


Рис. 3.13: Добавление нового ключа

3.6 Настройки подписи Git

Настраиваю автоматические подписи комментов git (рис. 3.14).

```
[tanya@tatyana-pavlova ~]$ git config --global user.signingkey C7E9E3BEEBA18E5B
[tanya@tatyana-pavlova ~]$ git config --global commit.gpgsign true
[tanya@tatyana-pavlova ~]$ git config --global gpg.program $(which gpg2)
[tanya@tatyana-pavlova ~]$
```

Рис. 3.14: Настройка подписей

3.7 Настройка gh

Начинаю авторизацию в gh (рис. 3.15).

```
[tanya@tatyana-pavlova ~]$ gh auth login
? Where do you use GitHub? 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
```

Рис. 3.15: Авторизация

Далее завершаю авторизацию на сайте и вижу сообщение о ее завершении (рис. 3.16).

```
✓ Authentication complete.
- gh config set -h github.com git_protocol https
✓ Configured git protocol
✓ Logged in as tanuha228
[tanya@tatyana-pavlova ~]$
```

Рис. 3.16: Сообщение о завершении авторизации

3.8 Создание репозитория курса на основе шаблона

Создаю директорию “Операционные системы” и создаю репозиторий на основе шаблона с помощью команды (рис. 3.17).

```
[tanya@tatyana-pavlova ~]$ mkdir -p "Операционные системы"
[tanya@tatyana-pavlova ~]$ ls
Видео  Загрузки  Музыка  Операционные  'Рабочий стол'  Шаблоны
Документы  Изображения  Общедоступные  'Операционные системы'  системы
[tanya@tatyana-pavlova ~]$ cd "Операционные системы"
[tanya@tatyana-pavlova Операционные системы]$ gh repo create study_2024-2025-os-intro-template yamadharma/course-directory-student-template-public
accepts at most 1 arg(s), received 2
```

Рис. 3.17: Создание директории и репозитория

Клонирую репозиторий (рис. 3.18).

```
[tanya@tatyana-pavlova Операционные системы]$ git clone --recursive https://github.com/tanuha228/study_2024-2025-os-intro.git os-intro
Клонирование в «os-intro»...
remote: Enumerating objects: 36, done.
remote: Counting objects: 100% (36/36), done.
remote: Compressing objects: 100% (35/35), done.
remote: Total 36 (delta 1), reused 21 (delta 0), pack-reused 0 (from 0)
Получение объектов: 100% (36/36), 19.37 KiB | 450.00 KiB/c, готово.
Определение изменений: 100% (1/1), готово.
```

Рис. 3.18: Клонирование

Удаляю лишние файлы и создаю необходимые каталоги (рис. 3.19).

```
CHANGELOG.md  config  course  LICENSE  Makefile  package.json
[tanya@tatyana-pavlova os-intro]$ rm package.json
[tanya@tatyana-pavlova os-intro]$ echo os-intro > COURSE
[tanya@tatyana-pavlova os-intro]$ make
Usage:
```

Рис. 3.19: Удаление и создание

Далее отправляю все новые вайлы на сервер.

4 Выводы

При выполнении данной лабораторной работы я узнала идеологию и применение средств контроля версий, освоила умение по работе с git.

Список литературы

1. лабораторная работа №2 [Электронный курс]