

# Лабораторная работа №2

Операционные системы

---

Куокконен Д. А.

19 февраля 2024

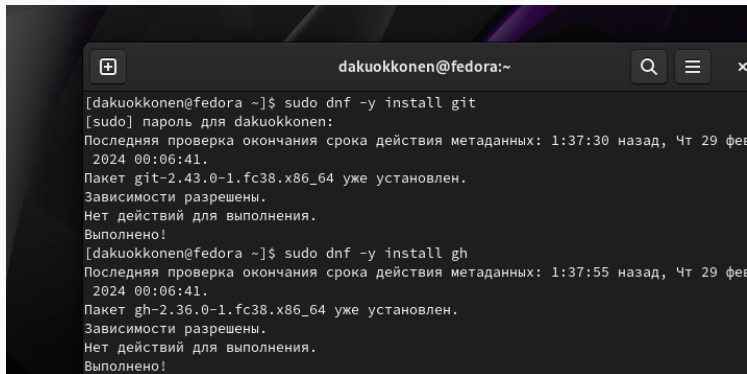
Российский университет дружбы народов, Москва, Россия

Цель данной лабораторной работы – изучение идеологии и применения средств контроля версий, освоение умения по работе с git.

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.

## Выполнение лабораторной работы. Установка ПО.

Я устанавливаю необходимое программное обеспечение *git* и *gh* через терминал с помощью команд: *dnf install git* и *dnf install gh* (рис. 1).

A screenshot of a terminal window titled 'dakuokkonen@fedora:~'. The window shows the execution of two commands: 'sudo dnf -y install git' and 'sudo dnf -y install gh'. The output for the first command shows that git-2.43.0-1.fc38.x86\_64 is already installed. The output for the second command shows that gh-2.36.0-1.fc38.x86\_64 is already installed. The terminal text is as follows:

```
[dakuokkonen@fedora ~]$ sudo dnf -y install git
[sudo] пароль для dakuokkonen:
Последняя проверка окончания срока действия метаданных: 1:37:30 назад, Чт 29 фев 2024 00:06:41.
Пакет git-2.43.0-1.fc38.x86_64 уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
[dakuokkonen@fedora ~]$ sudo dnf -y install gh
Последняя проверка окончания срока действия метаданных: 1:37:55 назад, Чт 29 фев 2024 00:06:41.
Пакет gh-2.36.0-1.fc38.x86_64 уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
```

Рис. 1: Установка git и gh

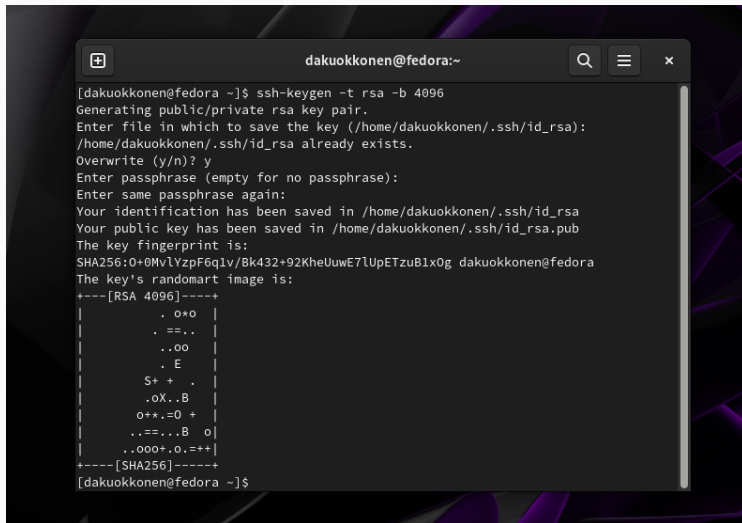
Задаю в качестве имени и email владельца репозитория свои имя, фамилию и электронную почту, настраиваю *utf-8* в выводе сообщений *git* для их корректного отображения, задаю имя начальной ветке, задаю параметры *autocrlf* и *safecrlf* для корректного отображения конца строки (рис. 2).

```
[dakuokkonen@fedora ~]$ git config --global user.name "<Darina Kuokkonen>"
[dakuokkonen@fedora ~]$ git config --global user.email "<1032230170@pfur.ru>"
[dakuokkonen@fedora ~]$ git config --global core.quotepath false
[dakuokkonen@fedora ~]$ git config --global init.default Branch master
[dakuokkonen@fedora ~]$ git config --global core.autocrlf input
[dakuokkonen@fedora ~]$ git config --global core.safecrlf warn
```

Рис. 2: Настройка конфига git

# Выполнение лабораторной работы. Создание ключа SSH

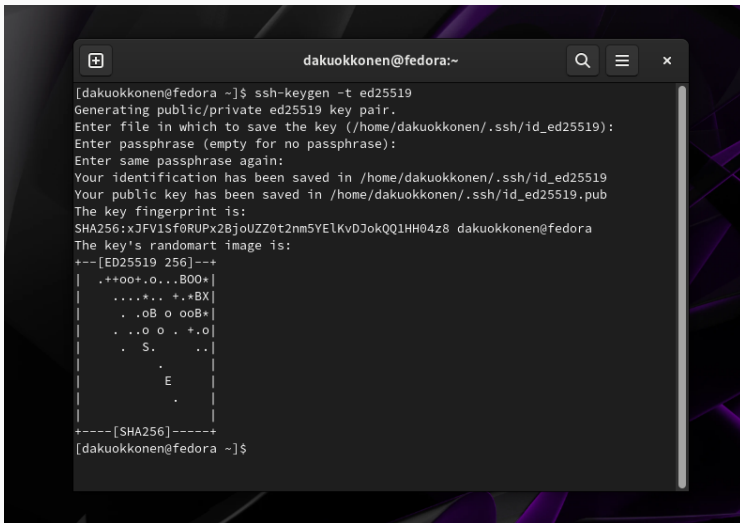
Создаю ключ ssh размером 4096 бит по алгоритму rsa (рис. 3).

A terminal window titled 'dakuokkonen@fedora:~' with search, menu, and close buttons. It shows the execution of 'ssh-keygen -t rsa -b 4096'. The process prompts for a file path (defaulting to /home/dakuokkonen/.ssh/id\_rsa), asks to overwrite (y/n), and prompts for a passphrase. It then displays the key's fingerprint (SHA256) and a randomart image for the RSA 4096 key. The terminal ends with the prompt '[dakuokkonen@fedora ~]\$'.

```
[dakuokkonen@fedora ~]$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/dakuokkonen/.ssh/id_rsa):
/home/dakuokkonen/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/dakuokkonen/.ssh/id_rsa
Your public key has been saved in /home/dakuokkonen/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:0+0MvLYzpF6q1v/Bk432+92KheUuwE7lUpETzuB1x0g dakuokkonen@fedora
The key's randomart image is:
+---[RSA 4096]-----+
|      . o+o  |
|      . ==.. |
|      ..oo   |
|      . E    |
|      S+ + .  |
|      .oX..B  |
|      o+*. =0 + |
|      ..==...B o|
|      ..ooo+.o.++|
+-----[SHA256]-----+
[dakuokkonen@fedora ~]$
```

## Выполнение лабораторной работы. Создание ключа SSH

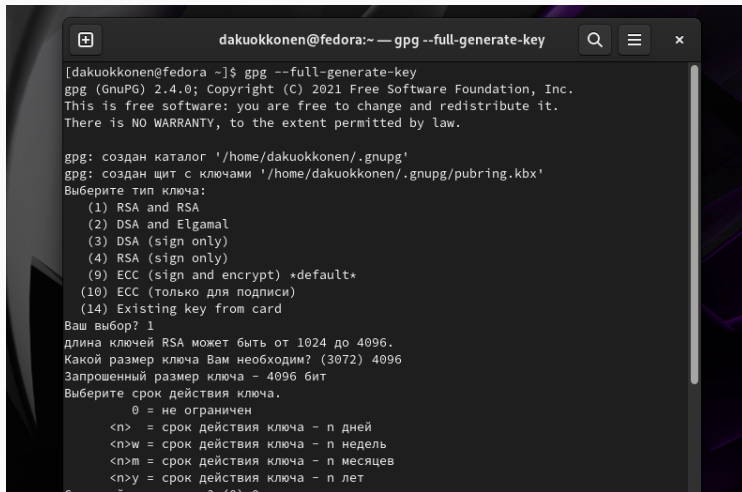
Создаю ключ ssh по алгоритму ed25519 (рис. 4).

A terminal window titled 'dakuokkonen@fedora:~' with search, menu, and close buttons. It shows the execution of 'ssh-keygen -t ed25519'. The user is prompted to enter a file path (defaulting to /home/dakuokkonen/.ssh/id\_ed25519), a passphrase (left empty), and to confirm the passphrase. The terminal then displays the key's fingerprint (SHA256) and a randomart image for the ED25519 key. The prompt returns to the shell.

```
[dakuokkonen@fedora ~]$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/dakuokkonen/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/dakuokkonen/.ssh/id_ed25519
Your public key has been saved in /home/dakuokkonen/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:xJFV1Sf0RUPx2Bj0UZZ0t2nm5YElKvDJokQQ1HH04z8 dakuokkonen@fedora
The key's randomart image is:
+--[ED25519 256]--+
|  .+oo+.o...B00*|
|    ....*..+.*BX|
|   .oB o ooB*|
|  . ..o o .+.o|
|   . S.    ..|
|           .|
|           E|
|           .|
+-----[SHA256]-----+
[dakuokkonen@fedora ~]$
```

## Выполнение лабораторной работы. Создание ключа GPG

Генерирую ключ *GPG*, затем выбираю тип ключа *RSA*, задаю максимальную длину ключа: 4096, выбираю неограниченный срок действия ключа. Завершаем настройку (рис. 5).

A screenshot of a terminal window titled 'dakuokkonen@fedora:~ — gpg --full-generate-key'. The terminal shows the execution of the 'gpg --full-generate-key' command. It displays the GPG version (2.4.0) and copyright information. It then prompts the user to select a key type from a list: (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), and (14) Existing key from card. The user selects option 1. Next, it prompts for the key size, showing the default is 4096 bits. Finally, it prompts for the expiration period, showing the default is 0 (no limit).

```
dakuokkonen@fedora:~ — gpg --full-generate-key
[dakuokkonen@fedora ~]$ gpg --full-generate-key
gpg (GnuPG) 2.4.0; Copyright (C) 2021 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

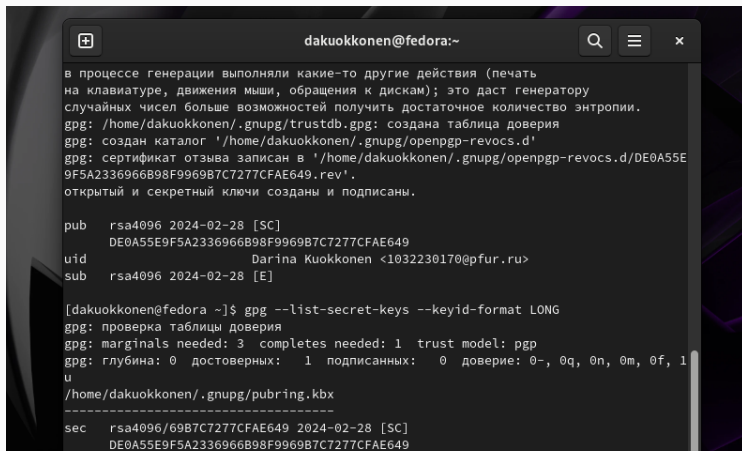
gpg: создан каталог '/home/dakuokkonen/.gnupg'
gpg: создан щит с ключами '/home/dakuokkonen/.gnupg/pubring.kbx'
Выберите тип ключа:
  (1) RSA and RSA
  (2) DSA and Elgamal
  (3) DSA (sign only)
  (4) RSA (sign only)
  (9) ECC (sign and encrypt) *default*
 (10) ECC (только для подписи)
 (14) Existing 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
```



Ранее у меня уже был создан аккаунт на Github, поэтому я просто захожу в свой аккаунт.

## Выполнение лабораторной работы. Добавление ключа GPG в Github

- Вывожу список созданных ключей в терминал
- Ищу в результате запроса отпечаток ключа
- Копирую его в буфер обмена, за это отвечает утилита *xclip* (рис. 6).



```
dakuokkonen@fedora:~  
в процессе генерации выполняли какие-то другие действия (печать  
на клавиатуре, движения мыши, обращения к дискам); это даст генератору  
случайных чисел больше возможностей получить достаточное количество энтропии.  
gpg: /home/dakuokkonen/.gnupg/trustdb.gpg: создана таблица доверия  
gpg: создан каталог '/home/dakuokkonen/.gnupg/openpgp-revocs.d'  
gpg: сертификат отзыва записан в '/home/dakuokkonen/.gnupg/openpgp-revocs.d/DE0A55E  
9F5A2336966B98F9969B7C7277CFAE649.rev'.  
открытый и секретный ключи созданы и подписаны.  
  
pub   rsa4096 2024-02-28 [SC]  
      DE0A55E9F5A2336966B98F9969B7C7277CFAE649  
uid     Darina Kuokkonen <1032230170@pfur.ru>  
sub   rsa4096 2024-02-28 [E]  
  
[dakuokkonen@fedora ~]$ 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, 1  
u  
/home/dakuokkonen/.gnupg/pubring.kbx  
-----  
sec   rsa4096/69B7C7277CFAE649 2024-02-28 [SC]  
      DE0A55E9F5A2336966B98F9969B7C7277CFAE649
```

Настраиваю автоматические подписи коммитов *git* (рис.7).

```
[dakuokkonen@fedora .gnupg]$ git config --global user.signingkey 69B7C7277CFAE649  
[dakuokkonen@fedora .gnupg]$ git config --global commit.gpgsign true  
[dakuokkonen@fedora .gnupg]$ git config --global gpg.program $(which gpg2)  
[dakuokkonen@fedora .gnupg]$
```

Рис. 7: Настройка подписей Git

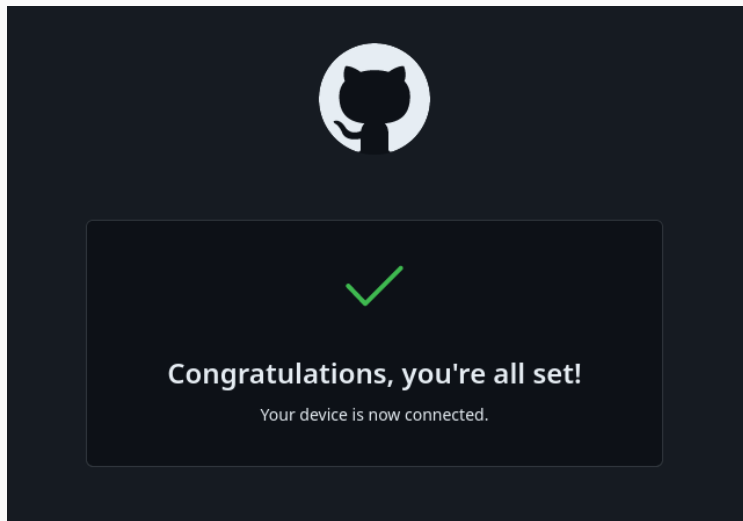
- Начинаю авторизацию в gh
- отвечаю на наводящие вопросы от утилиты
- выбираю авторизоваться через браузер (рис. 8).

```
[dakuokkonen@fedora .gnupg]$ gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Login with a web browser

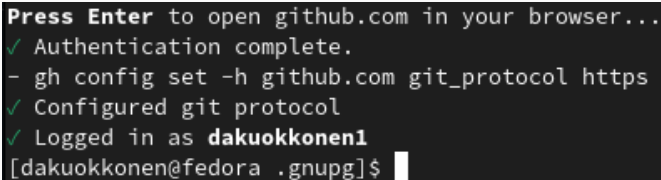
! First copy your one-time code: D753-A68C
Press Enter to open github.com in your browser... █
```

Рис. 8: Авторизация в gh

Завершаю авторизацию на сайте (рис. 9).



Видю сообщение о завершении авторизации (рис. 10).

A terminal window with a dark background and light-colored text. The text shows the completion of GitHub authentication steps: a prompt to press Enter, a success message, a command to set the git protocol to https, another success message, a login confirmation for user dakuokkonen1, and the final shell prompt.

```
Press Enter to open github.com in your browser...  
✓ Authentication complete.  
- gh config set -h github.com git_protocol https  
✓ Configured git protocol  
✓ Logged in as dakuokkonen1  
[dakuokkonen@fedora .gnupg]$
```

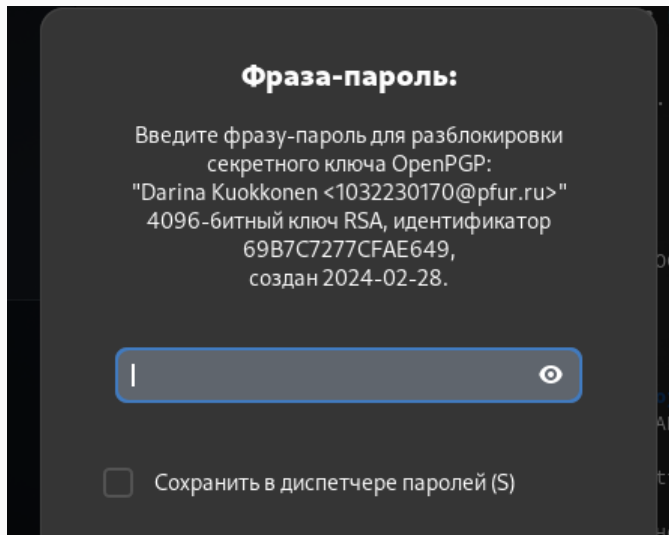
Рис. 10: Завершение авторизации

- Создаю директорию с помощью утилиты `mkdir` и флага `-p`
- Перехожу в только что созданную директорию “ОС”.
- В терминале ввожу команду `gh repo create study_2023-204_os-intro --template yamadharm/course-directory-student-trmplate --public`.
- После этого клонирую репозиторий к себе в директорию (рис. 11).

```
[dakuokkonen@fedora ОС]$ git clone --recursive https://github.com/dakuokkonen1/study_2023-2024_os-intro.git os-intro
```

Рис. 11: Создание репозитория

Ввожу фразу-пароль, установленную ранее (рис. 12).



**Фраза-пароль:**

Введите фразу-пароль для разблокировки  
секретного ключа OpenPGP:  
"Darina Kuokkonen <1032230170@pfur.ru>"  
4096-битный ключ RSA, идентификатор  
69B7C7277CFAE649,  
создан 2024-02-28.

☐ Сохранить в диспетчере паролей (S)

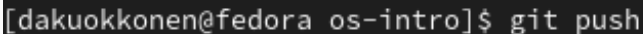


Перехожу в каталог курса с помощью утилиты `cd`, там я удаляю лишние файлы с помощью утилиты `rm`, далее создаю необходимые каталоги используя `make`. В конце концов добавляю все новые файлы для отправки на сервер (сохраняю добавленные изменения) с помощью команды `git add` и комментирую их с помощью `git commit` (рис. 13).

```
[dakuokkonen@fedora os-intro]$ git add .  
[dakuokkonen@fedora os-intro]$ git commit -am 'feat(main): make course structure'  
[master 3da57b0] feat(main): make course structure
```

Рис. 13: Работа с каталогами

Отправляю файлы на сервер с помощью *git push* (рис. 014).

A terminal window with a dark background. The prompt is [dakuokkonen@fedora os-intro]\$ and the command being entered is git push. The text is white on a black background.

```
[dakuokkonen@fedora os-intro]$ git push
```

Рис. 14: Отправка файлов на сервер

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

1. Лабораторная работа № 2 [Электронный ресурс] URL:  
<https://esystem.rudn.ru/mod/page/view.php?id=970819>