

Лабораторная работа №2

Операционные системы

Чувакина М. В.

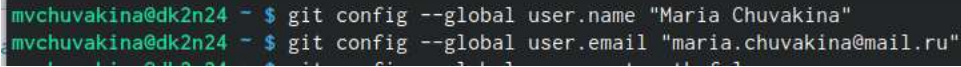
22 февраля 2024

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.

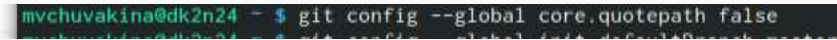
Задаю в качестве имени и email владельца репозитория свои имя, фамилию и электронную почту (рис. 1).

A terminal window with a dark background and light green text. It shows two lines of commands being executed. The first line sets the global user name to "Maria Chuvakina". The second line sets the global user email to "maria.chuvakina@mail.ru".

```
mvchuvakina@dk2n24 - $ git config --global user.name "Maria Chuvakina"  
mvchuvakina@dk2n24 - $ git config --global user.email "maria.chuvakina@mail.ru"
```

Рис. 1: Задаю имя и email владельца репозитория

Настраиваю utf-8 в выводе сообщений git для их корректного отображения (рис. 2).



```
mvchuvakina@dk2n24 ~$ git config --global core.quotePath false
mvchuvakina@dk2n24 ~$ git config --global init.defaultBranch master
```

Рис. 2: Настройка utf-8 в выводе сообщений git

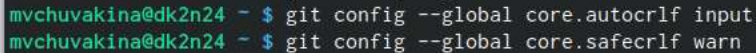
Начальной ветке задаю имя master (рис. 3).

A terminal window with a dark background. The prompt is 'mvchuvakina@dk2n24 ~ \$'. The command entered is 'git config --global init.defaultBranch master'.

```
mvchuvakina@dk2n24 ~ $ git config --global init.defaultBranch master
```

Рис. 3: Задаю имя начальной ветки

Задаю параметры autocrlf и safecrlf (рис. 4).

A terminal window with a dark background and light green text. It shows two lines of commands being entered at a prompt. The first line sets 'core.autocrlf' to 'input', and the second line sets 'core.safecrlf' to 'warn'.

```
mvchuvakina@dk2n24 - $ git config --global core.autocrlf input  
mvchuvakina@dk2n24 - $ git config --global core.safecrlf warn
```

Рис. 4: Задаю параметры autocrlf и safecrlf

Выполнение лабораторной работы. Создание ключа SSH

Создаю ключ ssh размером 4096 бит по алгоритму rsa (рис. 5).

```
mvchuvakina@dk2n24 ~$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/afs/.dk.sci.pfu.edu.ru/home/m/v/mvchuvakina/.ssh/id_rsa):
/afs/.dk.sci.pfu.edu.ru/home/m/v/mvchuvakina/.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 /afs/.dk.sci.pfu.edu.ru/home/m/v/mvchuvakina/.ssh/id_rsa
Your public key has been saved in /afs/.dk.sci.pfu.edu.ru/home/m/v/mvchuvakina/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:K0Sxq951oAVeYyJqha5ZcE2o7vd5qJMDrx5wGCOdqEg mvchuvakina@dk2n24
The key's randomart image is:
+----[RSA 4096]-----+
|  .+ . |
| .+ .+ . |
|..o+.+. |
|o+.+.+. |
|OEB+.S |
|B# o.o.o |
|X+ .o.o |
|++o.o.o |
|==o. . |
+----[SHA256]-----+
mvchuvakina@dk2n24 ~$
```

Рис. 5: Генерация ssh ключа по алгоритму rsa

Выполнение лабораторной работы. Создание ключа SSH

Создаю ключ ssh по алгоритму ed25519 (рис. 6).

```
+-----[SHA256]-----+
mvchuvakina@dk2n24 ~$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/afs/.dk.sci.pfu.edu.ru/home/n/v/mvchuvakina/.ssh/id_ed25519):
/afs/.dk.sci.pfu.edu.ru/home/n/v/mvchuvakina/.ssh/id_ed25519 already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /afs/.dk.sci.pfu.edu.ru/home/n/v/mvchuvakina/.ssh/id_ed25519
Your public key has been saved in /afs/.dk.sci.pfu.edu.ru/home/n/v/mvchuvakina/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:FYpxJ9QGV23+aEhpymzjp1C8W4nk9x005s/LPLR0Q mvchuvakina@dk2n24
The key's randomart image is:
+---[ED25519 256]---+
|
|      .
|      o
|     o o
|    .o+ .
|   oE+oB.+
|  .oo=ooB o
| .oo+ .o
|  +B+oo
|  +X=.o
|
+-----[SHA256]-----+
mvchuvakina@dk2n24 ~$
```

Рис. 6: Генерация ssh ключа по алгоритму ed25519

Выполнение лабораторной работы. Создание ключа GPG

Генерирую ключ GPG, затем выбираю тип ключа RSA and RSA, задаю максимальную длину ключа: 4096, оставляю неограниченный срок действия ключа. Далее отвечаю на вопросы программы о личной информации (рис. 7).

```
stus@stuskladmail:~$ gpg --full-generate-key
gpg: (GnuPG) 2.3.42: 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.

Выберите тип ключа:
  (1) RSA и RSA (по умолчанию)
  (2) RSA и ElGamal
  (3) RSA (только для подписи)
  (4) RSA (только для шифрования)
  (14) Прямой на карте ключ
Ваш выбор? 1
Длина ключей RSA может быть от 1024 до 4096.
Какой размер ключа Вам необходим? (3072) 4096
Запланированный размер ключа - 4096 бит
Выберите срок действия ключа...
  0 = не ограничен
  1y  = срок действия ключа - 1 дней
  1mo  = срок действия ключа - 1 месяц
  1y   = срок действия ключа - 1 год
  10y  = срок действия ключа - 10 лет
Срок действия ключа? (0) 0
Срок действия ключа не ограничен
Все верно? (y/N) y

GnuPG должен составить идентификатор пользователя для идентификации ключа.

Ваше полное имя: Мария
Адрес электронной почты: maria.stusklad@mail.ru
Описание:
Используйте таблица значений 'utf-8'.
Вы выбрали следующий идентификатор пользователя:
"Мария <maria.stusklad@mail.ru>"

Создать (Y)Нет, (C)Принимать, (E)Адрес, (O)Принять (Q)Выход? 0
```


Ввожу в терминале команду, с помощью которой копирую сам ключ GPG в буфер обмена (рис. 9).

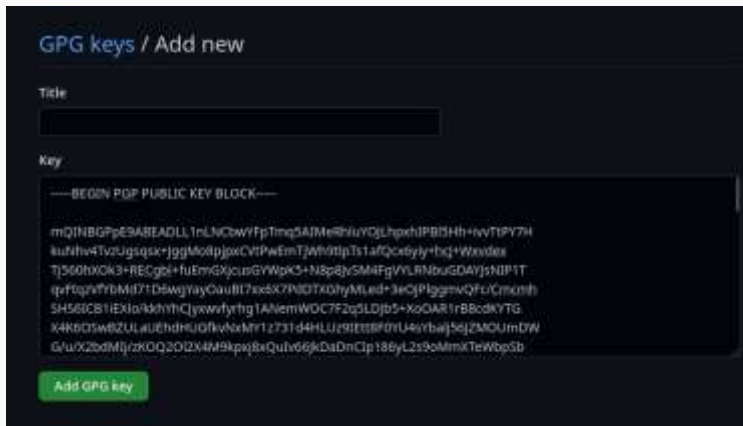


```
avchuvakina@dk2n24 ~$ gpg --armor --export <PGP fingerprint> | xclip -sel clip
bash: синтаксическая ошибка рядом с неожиданным маркером «|»
avchuvakina@dk2n24 ~$ gpg --armor --export 7E6C02CF7752B654 | xclip -sel clip
avchuvakina@dk2n24 ~$
```

Рис. 9: Копирование ключа в буфер обмена

Выполнение лабораторной работы. Добавление ключа GPG в Github

- Открываю настройки GitHub, ищу среди них добавление GPG ключа.
- Нажимаю на “New GPG key” и вставляю в поле ключ из буфера обмена (рис. 10).
- Я добавила ключ GPG на GitHub.



GPG keys / Add new

Title

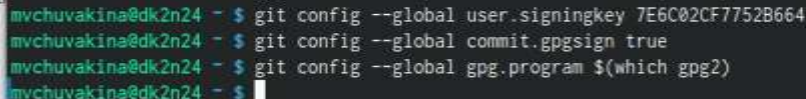
Key

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

mQINBGFPeE9A8EADLL1nLnCwYfFpTmq5AJMeRhuYQJLhpchIPB5Hh+vvT1PY7H
ku4hv4Tv2Ugsqsk+JggMo8pJpxCvtPwEmTjWh9tlpTstafQcx6yly+hcj+Wxvdx
Tj560hXCK3+RECgbl+fuEmGXjousGYWpK5+N8p8JvSM4FgVYLRNbuGDAYjsNIP1T
qrFtqzvYrBMD71D6wjYayCaull7xx6X7PNDTko0hyMled+3eCjPiggewQfuCm0th
SH56ICB1iEXioKdhYhCjYxwvfyrtg1ANemWOC7F2q5LDJb5+XoQAR1rBBcdKYTG
X4K6QSw6ZULau6hdhuGfkwXmT1z731d4HLUz91E6Bf0rU4yrbaj56jZMOUmDW
G/u/X2bdMj/zK0Q20IZX4M9kpvj8xQuIv66kDaDnClp186yL2s9oMmXTeWbpSb

Add GPG key

Настраиваю автоматические подписи коммитов git (рис. 11).

A terminal window with a dark background and green text. It shows four lines of commands being entered at a prompt. The first line sets the global signing key. The second line enables commit signing. The third line sets the GPG program to the one found by 'which'. The fourth line shows the prompt again with a cursor.

```
mvchuvakina@dk2n24 - $ git config --global user.signingkey 7E6C02CF7752B664
mvchuvakina@dk2n24 - $ git config --global commit.gpgsign true
mvchuvakina@dk2n24 - $ git config --global gpg.program $(which gpg2)
mvchuvakina@dk2n24 - $
```

Рис. 11: Настройка подписей Git

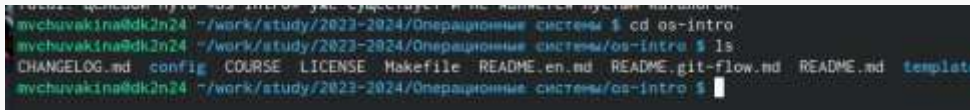
- Начинаю авторизацию в gh
- отвечаю на наводящие вопросы от утилиты
- выбираю авторизоваться через браузер (рис. 12).

```
mychurkin@mailbox2424 ~ /workstudy/2023-2024/Операционные системы $ cd
mychurkin@mailbox2424 ~ $ 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: EE61-F5AC
  Press Enter to open github.com in your browser...
[14179:14235:6214/185100.176836:ERROR:nss_util.cc(357)] After loading Root Certs, loaded==false: NSS error code: -8018
[14179:14179:6214/185111.140145:ERROR:object_proxy.cc(576)] Failed to call method: org.freedesktop.ScreenSaver.GetActive: object_path= /org/freedesktop/ScreenSaver: org.freedesktop.DBus.Error.NotImplemented: A
is method is not part of the idle inhibition specification: https://specifications.freedesktop.org/idle-
inhibit-spec/latest/
[14179:14179:6214/185111.116254:ERROR:object_proxy.cc(576)] Failed to call method: org.gnome.ScreenSaver
.GetActive: object_path= /org/gnome/ScreenSaver: org.freedesktop.DBus.Error.ServiceUnknown: GDBus error
org.freedesktop.DBus.Error.ServiceUnknown: The name org.gnome.Shell.ScreenSaver was not provided by a
service files
- Authentication complete.
- gh config set -h github.com git.protocol https
  Configured git protocol
```

Рис. 12: Авторизация в gh

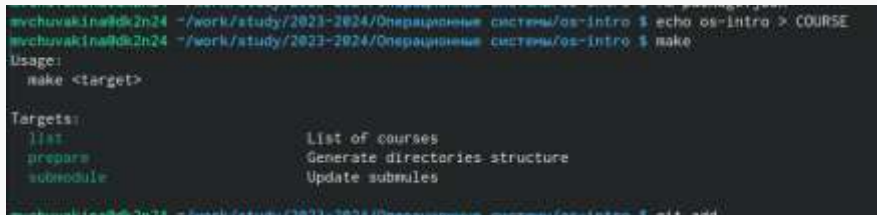
- Перехожу в каталог курса
- Проверяю содержание каталога (рис. 14).

A terminal window with a dark background and light blue/green text. The prompt is 'nvchuvakina@dk2n24'. The first command is 'cd os-intro', and the second is 'ls'. The output of 'ls' shows a list of files: 'CHANGELOG.md', 'config', 'COURSE', 'LICENSE', 'Makefile', 'README.en.md', 'README.git-flow.md', 'README.md', and 'template'.

```
nvchuvakina@dk2n24 ~/work/study/2023-2024/Операционные системы $ cd os-intro
nvchuvakina@dk2n24 ~/work/study/2023-2024/Операционные системы/os-intro $ ls
CHANGELOG.md  config  COURSE  LICENSE  Makefile  README.en.md  README.git-flow.md  README.md  template
```

Рис. 14: Перемещение между директориями

- Удаляю лишние файлы
- Создаю необходимые каталоги, используя makefile (рис. 15).



```
mvchuvakina@dk2n24 ~/work/study/2023-2024/Операционные системы/os-intro $ echo os-intro > COURSE
mvchuvakina@dk2n24 ~/work/study/2023-2024/Операционные системы/os-intro $ make
Usage:
  make <target>

Targets:
  list          List of courses
  prepare       Generate directories structure
  submodule     Update submodules
mvchuvakina@dk2n24 ~/work/study/2023-2024/Операционные системы/os-intro $ git add
```

Рис. 15: Удаление файлов и создание каталогов

- Сохраняю добавленные изменения
- Комментирую их с помощью git commit (рис. 16).

```
svchuvakina@dk2n24 ~/work/study/2023-2024/Операционные системы/os-intro $ git add .
svchuvakina@dk2n24 ~/work/study/2023-2024/Операционные системы/os-intro $ git commit -m 'feat(main): make
ke course structure'
[master d88e8ba] feat(main): make course structure
 2 files changed, 1 insertion(+), 14 deletions(-)
 delete mode 100644 package.json
svchuvakina@dk2n24 ~/work/study/2023-2024/Операционные системы/os-intro $ git push
Перечисление объектов: 5, готово.
Подсчет объектов: 100% (5/5), готово.
При сжатии изменений используется до 5 потоков
Сжатие объектов: 100% (2/2), готово.
Запись объектов: 100% (3/3), 285 байтов | 285.00 КиБ/с, готово.
Всего 3 (изменений 1), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/svchuvakina/study_2023-2024_os-intro.git
   0e1f392..d88e8ba  master -> master
svchuvakina@dk2n24 ~/work/study/2023-2024/Операционные системы/os-intro $
```

Рис. 16: 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>