

Лабораторная работа 2

Первоначальная настройка git

Богданюк А.В

Российский университет дружбы народов, Москва, Россия

Вводная часть

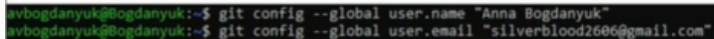
- Изучить идеологию и применение средств контроля версий. Освоить умения по работе с git.

1. Базовая настройка git
2. Создание ssh ключа
3. Верификация коммитов с помощью PGP
4. Проверка коммитов в Git
5. Шаблон рабочего пространства

Системы контроля версий (Version Control System, VCS) применяются при работе нескольких человек над одним проектом. Обычно основное дерево проекта хранится в локальном или удалённом репозитории, к которому настроен доступ для участников проекта. При внесении изменений в содержание проекта система контроля версий позволяет их фиксировать, совмещать изменения, произведённые разными участниками проекта, производить откат к любой более ранней версии проекта, если это требуется. В классических системах контроля версий используется централизованная модель, предполагающая наличие единого репозитория для хранения файлов. Выполнение большинства функций по управлению версиями осуществляется специальным сервером. Участник проекта (пользователь) перед началом работы посредством определённых команд получает нужную ему версию файлов. После внесения изменений, пользователь размещает новую версию в хранилище. При этом предыдущие версии не удаляются из центрального хранилища и к ним можно вернуться в любой момент. Сервер может сохранять не полную версию изменённых файлов, а производить так называемую дельта-компрессию — сохранять только изменения между

Основная часть

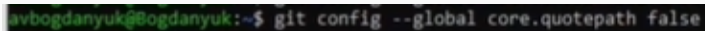
Задаю имя и email владельца репозитория (рис. 1).

A terminal window with a dark background and light-colored text. It shows two lines of commands being executed. The first line sets the global user name to 'Anna Bogdanyuk', and the second line sets the global user email to 'silverblood2606@gmail.com'. Both prompts are 'avbogdanyuk@bogdanyuk:~\$'.

```
avbogdanyuk@bogdanyuk:~$ git config --global user.name "Anna Bogdanyuk"  
avbogdanyuk@bogdanyuk:~$ git config --global user.email "silverblood2606@gmail.com"
```

Рис. 1: Имя и email владельца репозитория

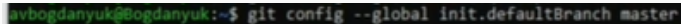
Настраиваю utf-8 в выводе сообщений git (рис. 2).



```
lavbogdanyuk@bogdanyuk:~$ git config --global core.quotePath false
```

Рис. 2: utf-8

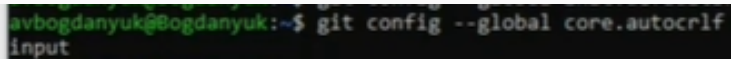
Настраиваю верификацию и подписание коммитов git. Задаю имя начальной ветки (master) (рис. 3).

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

```
avbogdanyuk@bogdanyuk:~$ git config --global init.defaultBranch master
```

Рис. 3: Имя начальной ветки

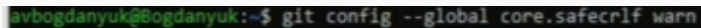
Устанавливаю параметр autocrlf (рис. 4).

A terminal window with a black background and green text. The prompt is 'avbogdanyuk@bogdanyuk:~\$'. The command entered is 'git config --global core.autocrlf input'.

```
avbogdanyuk@bogdanyuk:~$ git config --global core.autocrlf  
input
```

Рис. 4: autocrlf

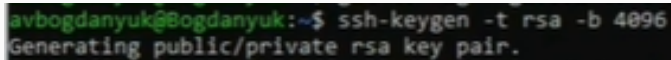
Устанавливаю параметр safecrlf (рис. 5).

A terminal window with a dark background. The prompt is 'avbogdanyuk@Bogdanyuk:~\$'. The command entered is 'git config --global core.safecrlf warn'.

```
avbogdanyuk@Bogdanyuk:~$ git config --global core.safecrlf warn
```

Рис. 5: safecrlf

Создаю ключ ssh по алгоритму rsa с ключём размером 4096 бит (рис. 6).



```
avbogdanyuk@Bogdanyuk:~$ ssh-keygen -t rsa -b 4096  
Generating public/private rsa key pair.
```

Рис. 6: Алгоритм rsa

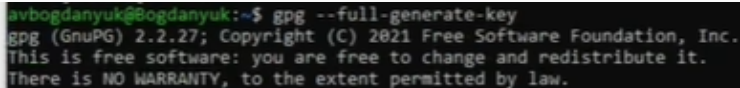
Выполнение лабораторной работы

По алгоритму ed25519 (рис. 7).

```
avbogdanyuk@Bogdanyuk:~$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/avbogdanyuk/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/avbogdanyuk/.ssh/id_ed25519
Your public key has been saved in /home/avbogdanyuk/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:Ar3Mxr8kVn1Ibrqf4EMh5eL8YuMh2ta/3U1UvJ+H5WU avbogdanyuk@Bogdanyuk
The key's randomart image is:
+--[ED25519 256]--+
|
|   . . .
|  . . O . . O
|   =O.O+ . . .
|  oBoS.= . . E
|   .O+.O . . +=
|  ..+O= . . O.+
|   O.O==O= O O .
|  ...O.+*++ . .
+----[SHA256]-----+
```

Рис. 7: Алгоритм ed25519

Генерируем ключ pgp (рис. 8).

A terminal window with a black background and green text. The prompt is 'avbogdanyuk@Bogdanyuk:~\$'. The command entered is 'gpg --full-generate-key'. The output shows the GPG version (2.2.27), copyright information (© 2021 Free Software Foundation, Inc.), and a disclaimer: 'This is free software: you are free to change and redistribute it. There is NO WARRANTY, to the extent permitted by law.'

```
avbogdanyuk@Bogdanyuk:~$ gpg --full-generate-key
gpg (GnuPG) 2.2.27; 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.
```

Рис. 8: Ключ pgp

Ввожу passphrase, чтобы защитить мой ключ, учетная запись на github осталась с прошлого семестра (рис. 9).

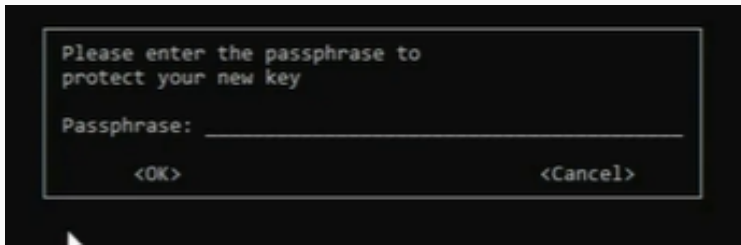
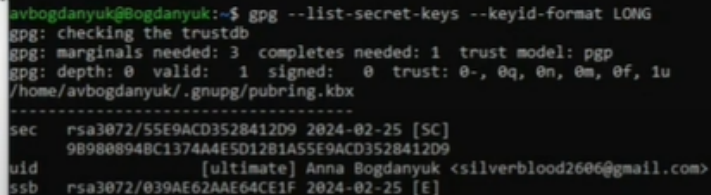


Рис. 9: Passphrase

Вывожу список ключей и копирую отпечаток приватного ключа (рис. 10).

A terminal window with a black background and green text. The command 'gpg --list-secret-keys --keyid-format LONG' is entered. The output shows GPG status messages and a list of keys. The first key is a secret key (sec) with ID 'rsa3072/55E9ACD3528412D9', expiration '2024-02-25', and protection '[SC]'. The second line shows the fingerprint '9B980894BC1374A4ESD12B1A55E9ACD3528412D9'. The third line shows the user ID '[ultimate] Anna Bogdanyuk <silverblood2606@gmail.com>'. The fourth line is the private key (ssb) with ID 'rsa3072/039AE62AAE64CE1F', expiration '2024-02-25', and protection '[E]'.

```
avbogdanyuk@Bogdanyuk:~$ gpg --list-secret-keys --keyid-format LONG
gpg: checking the trustdb
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
/home/avbogdanyuk/.gnupg/pubring.kbx
-----
sec   rsa3072/55E9ACD3528412D9 2024-02-25 [SC]
      9B980894BC1374A4ESD12B1A55E9ACD3528412D9
uid           [ultimate] Anna Bogdanyuk <silverblood2606@gmail.com>
ssb   rsa3072/039AE62AAE64CE1F 2024-02-25 [E]
```

Рис. 10: Список ключей

Копирую сгенерированный PGP ключ в буфер обмена (рис. 11).

```
avbogdanyuk@Bogdanyuk:~$ gpg --armor --export 55E9ACD3528412D9 | xclip -s
```

Рис. 11: Копирование ключа

Перехожу в настройки Github, нажимаю на кнопку New GPG key и вставляю полученный ключ в поле ввода (рис. 12).

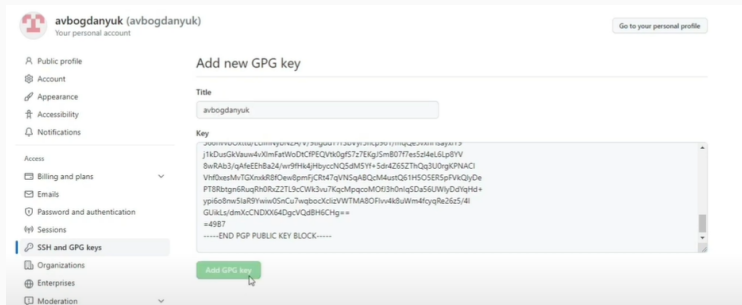
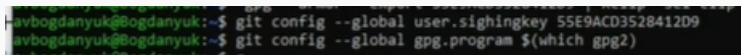


Рис. 12: New GPG key

Используя введенных email, указываю Git применять его при подписи коммитов (рис. 13).

A terminal window with a dark background and light-colored text. The prompt is 'avbogdanyuk@Bogdanyuk:~\$'. The first command is 'git config --global user.signingkey 55E9ACD3528412D9'. The second command is 'git config --global gpg.program \$(which gpg2)'.

```
avbogdanyuk@Bogdanyuk:~$ git config --global user.signingkey 55E9ACD3528412D9
avbogdanyuk@Bogdanyuk:~$ git config --global gpg.program $(which gpg2)
```

Рис. 13: Подпись коммитов

Устанавливаю gh (рис. 14).

A terminal window with a black background and green text. The prompt is 'lavbogdanyuk@bogdanyuk:~\$'. The command entered is 'sudo snap install gh'. The output is '# version 2.6.0-15-g1a10fd5a' followed by a new line 'Setup snap "gh" (502) security profiles'.

```
lavbogdanyuk@bogdanyuk:~$ sudo snap install gh      # version 2.6.0-15-g1a10fd5a
Setup snap "gh" (502) security profiles
```

Рис. 14: gh

Авторизуюсь через браузер (рис. 15).

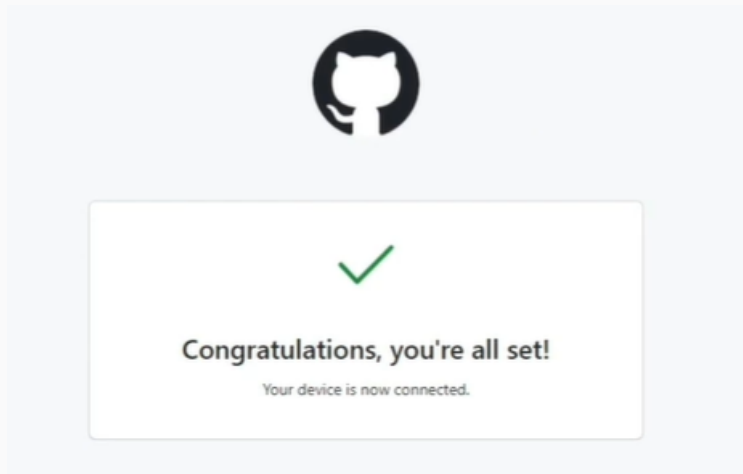


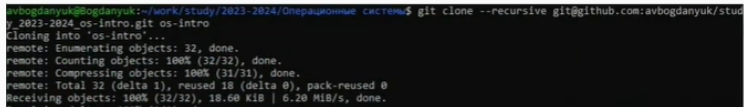
Рис. 15: Настройка gh

Создаю репозиторий курса на основе шаблона (рис. 16).

```
avbogdanyuk@Bogdanyuk:~$ mkdir -p ~/work/study/2023-2024/"Операционные системы"
avbogdanyuk@Bogdanyuk:~$ cd ~/work/study/2023-2024/"Операционные системы"
avbogdanyuk@Bogdanyuk:~/work/study/2023-2024/Операционные системы$ gh repo create study_2023-2024_os-intro --template=yamadharma/course-directory-student-template --public
2024/02/25 20:14:15.912000 cmd_run.go:1055: WARNING: cannot start document portal: dial unix /run/user/1000/bus: connect
: no such file or directory
✔ Created repository avbogdanyuk/study_2023-2024_os-intro on GitHub
```

Рис. 16: Создание репозитория


Клонирую репозиторий с шаблона (рис. 17).



```
avbogdanyuk@Bogdanyuk:~/work/study/2023-2024/Операционные системы$ git clone --recursive git@github.com:avbogdanyuk/study_2023-2024_os-intro.git os-intro
Cloning into 'os-intro'...
remote: Enumerating objects: 32, done.
remote: Counting objects: 100% (32/32), done.
remote: Compressing objects: 100% (31/31), done.
remote: Total 32 (delta 1), reused 18 (delta 0), pack-reused 0
Receiving objects: 100% (32/32), 18.60 KiB | 6.20 MiB/s, done.
```

Рис. 17: Клонирование репозитория

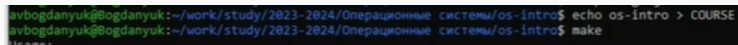
Перехожу в каталог курса и удаляю лишний файл (рис. 18).

A terminal window with a black background and green text. The first line shows the user 'avbogdanyuk' at host 'Bogdanyuk' in the directory '~/work/study/2023-2024/Операционные системы', executing the command 'cd os-intro'. The second line shows the same user and host in the directory '~/work/study/2023-2024/Операционные системы/os-intro', executing the command 'rm package.json'.

```
avbogdanyuk@Bogdanyuk:~/work/study/2023-2024/Операционные системы$ cd os-intro
avbogdanyuk@Bogdanyuk:~/work/study/2023-2024/Операционные системы/os-intro$ rm package.json
```

Рис. 18: Удаление файла

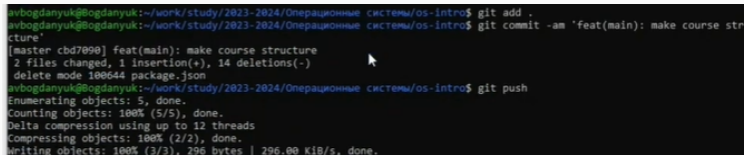
Создаю необходимые каталоги (рис. 19).

A terminal window with a black background and green text. The prompt is 'avbogdanyuk@Bogdanyuk:~/work/study/2023-2024/Операционные системы/os-intro\$'. The first command is 'echo os-intro > COURSE' and the second is 'make'.

```
avbogdanyuk@Bogdanyuk:~/work/study/2023-2024/Операционные системы/os-intro$ echo os-intro > COURSE
avbogdanyuk@Bogdanyuk:~/work/study/2023-2024/Операционные системы/os-intro$ make
```

Рис. 19: Создание каталогов

Отправляю файлы на сервер (рис. 20).



```
avbogdanyuk@Bogdanyuk:~/work/study/2023-2024/Операционные системы/os-intro$ git add .
avbogdanyuk@Bogdanyuk:~/work/study/2023-2024/Операционные системы/os-intro$ git commit -am 'feat(main): make course structure'
[master cbd7090] feat(main): make course structure
 2 files changed, 1 insertion(+), 14 deletions(-)
 delete mode 100644 package.json
avbogdanyuk@Bogdanyuk:~/work/study/2023-2024/Операционные системы/os-intro$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 296 bytes | 296.00 KiB/s, done.
```

Рис. 20: Отправление файлов на сервер

Заключительная часть

В ходе выполнения лабораторной работы были изучены идеология и применение средств контроля версий и были освоены умения по работе с git.