Лабораторная работы 2

Первоначальная настройка git

Богданюк Анна Васильевна

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

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

2 Задание

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

3 Теоретическое введение

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 последовательными версиями, что позволяет уменьшить объём хранимых данных. Системы контроля версий поддерживают возможность отслеживания и разрешения конфликтов, которые могут возникнуть при работе нескольких человек над одним файлом. Можно объединить (слить) изменения, сделанные разными участниками (автоматически или вручную), вручную выбрать нужную версию, отменить изменения вовсе или заблокировать файлы для изменения. В зави-

симости от настроек блокировка не позволяет другим пользователям получить рабочую копию или препятствует изменению рабочей копии файла средствами файловой системы ОС, обеспечивая таким образом, привилегированный доступ только одному пользователю, работающему с файлом.

4 Выполнение лабораторной работы

Задаю имя и email владельца репозитория (рис. 4.1).

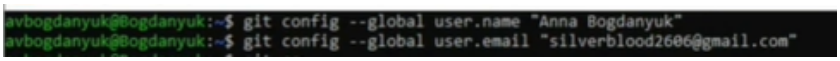

Рис. 4.1: Имя и email владельца репозитория

Настраиваю utf-8 в выводе сообщений git (рис. 4.2).

Рис. 4.2: utf-8

Настраиваю верификацию и подписание коммитов git. Задаю имя начальной ветки (master) (рис. 4.3).

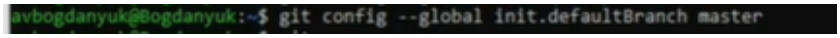

Рис. 4.3: Имя начальной ветки

Устанавливаю параметр autocrlf (рис. 4.4).

Рис. 4.4: autocrlf

Устанавливаю параметр safecrlf (рис. 4.5).

Рис. 4.5: safecrlf

Создаю ключ ssh по алгоритму rsa с ключём размером 4096 бит (рис. 4.6).

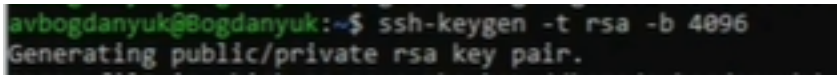

Рис. 4.6: Алгоритм rsa

По алгоритму ed25519 (рис. 4.7).

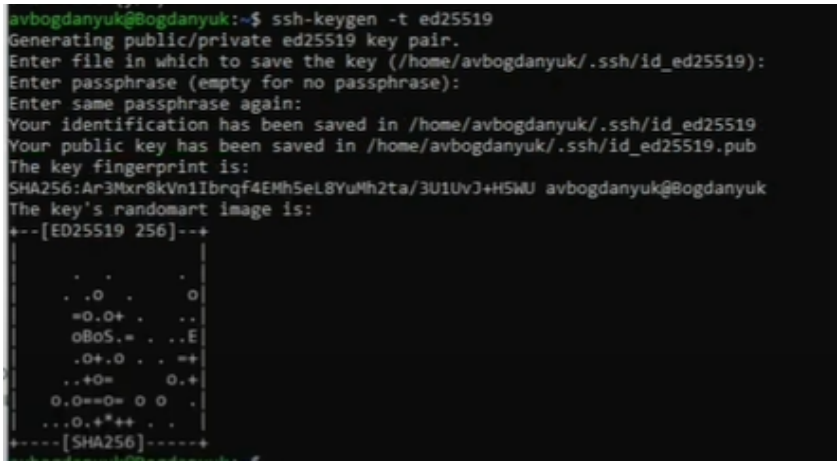

Рис. 4.7: Алгоритс ed25519

Генерируем ключ pgp (рис. 4.8).

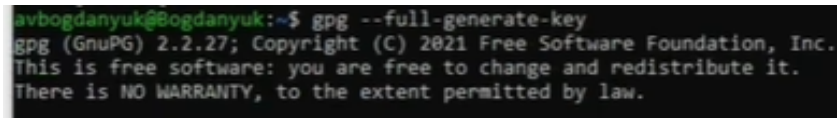

Рис. 4.8: Ключ pgp

Ввожу passphrase, чтобы защитить мой ключ, учетная запись на github осталась с прошлого семестра (рис. 4.9).

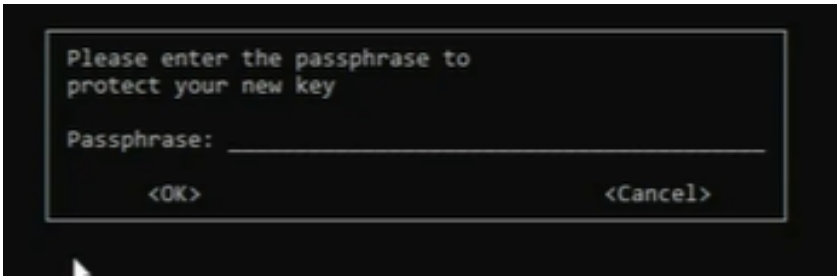

Рис. 4.9: Passphrase

Вывожу спислк ключей и копирую отпечаток приватного ключа (рис. 4.10).

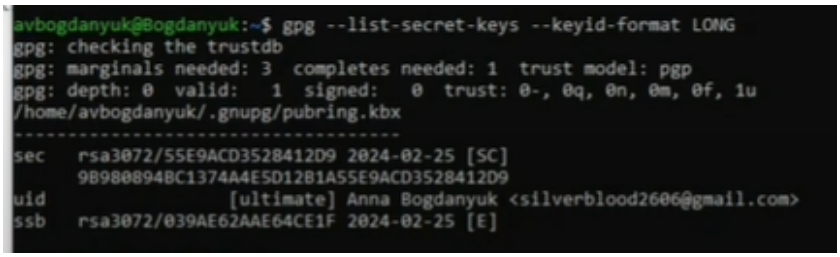

Рис. 4.10: Список ключей

Копирую сгенерированный PGP ключ в буфер обмена (рис. 4.11).

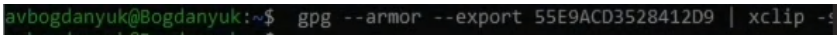

Рис. 4.11: Копирование ключа

Перехожу в настройки Github, нажимаю на кнопку New GPG key и вставляю полученный ключ в поле ввода (рис. 4.12).

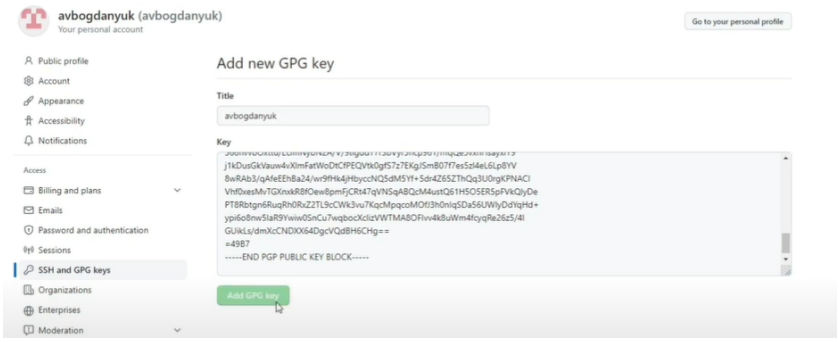

Рис. 4.12: New GPG key

Использую введённых email, указываю Git применять его при подписи коммитов (рис. 4.13).

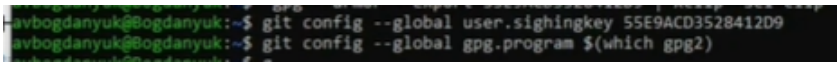

Рис. 4.13: Подпись коммитов

Устанавливаю gh (рис. 4.14).

Рис. 4.14: gh

Авторизуюсь через браузер (рис. 4.15).

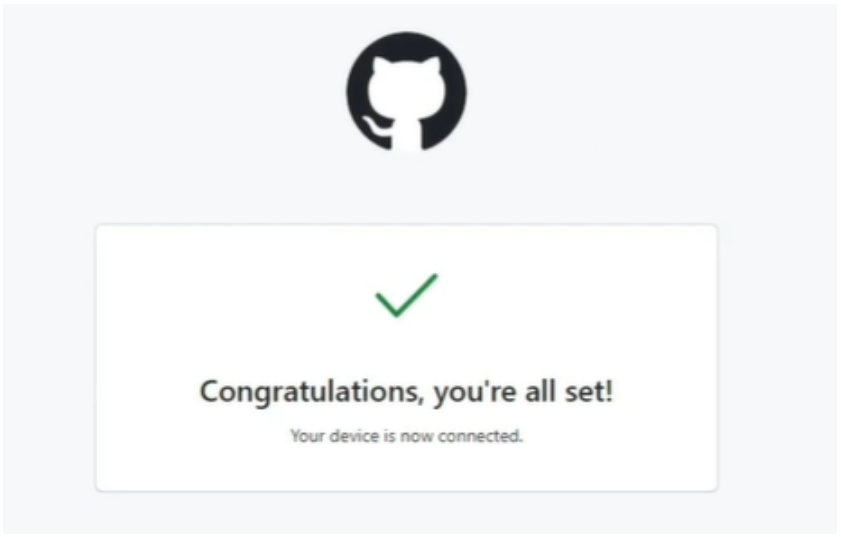

Рис. 4.15: Настройка gh

Создаю репозироторий курса на основе шаблона (рис. 4.16).

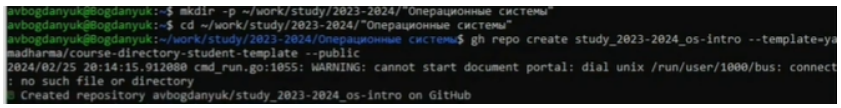

Рис. 4.16: Создание репозитория

Клонирую репозиторий с шаблона (рис. 4.17).

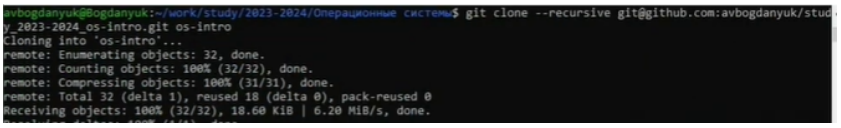

Рис. 4.17: Клонирование репозитория

Перехожу в каталог курса и удаляю лишний файл (рис. 4.18).

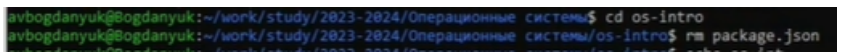

Рис. 4.18: Удаление файла

Создаю необходимые каталоги (рис. 4.19).

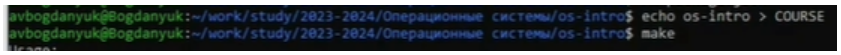

Рис. 4.19: Создание каталогов

Отправляю файлы на сервер (рис. 4.20).

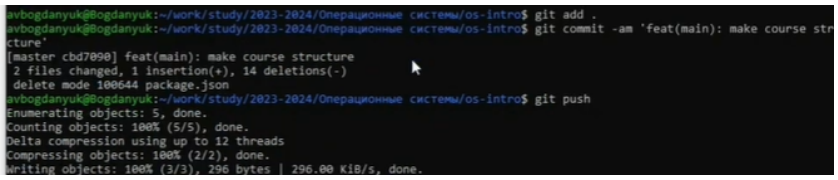

Рис. 4.20: Отправление файлов на сервер

5 Выводы

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