Отчёт по лабораторной работе №2

Операционные системы

Алина Андреевна Кайнова

Содержание

# 1 Цель работы

Изучить идеологию и применение средств контроля версий, приобрести навыки работы с git

# 2 Задание

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

# 3 Теоретическое введение

Системы контроля версий (Version Control System, VCS) применяются при работе нескольких человек над одним проектом. 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 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 изменения, сделанные разными участниками (автоматически или вручную), вручную выбрать нужную версию, отменить изменения вовсе или заблокировать файлы для изменения. В зависимости 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. Системы контроля версий также могут обеспечивать дополнительные, более гибкие функциональные возможности. Например, они могут поддерживать работу с несколькими версиями одного файла, сохраняя общую историю изменений до точки ветвления версий и собственные истории изменений каждой ветви. Кроме того, обычно доступна информация о том, кто из участников, когда и какие изменения вносил. Обычно такого рода информация хранится в журнале изменений, доступ к которому можно ограничить.В отличие от классических, в распределённых системах контроля версий центральный репозиторий не является обязательным.Среди классических VCS наиболее известны CVS, Subversion, а среди распределённых — Git, Bazaar, Mercurial. Принципы их работы схожи, отличаются они в основном синтаксисом используемых в работе команд. Система контроля версий Git представляет собой набор программ командной строки. Доступ к ним можно получить из терминала посредством ввода команды git с различными опциями.Благодаря тому, что Git является распределённой системой контроля версий, резервную копию локального хранилища можно сделать простым копированием или архивацией.

# 4 Выполнение лабораторной работы

## 4.1 Создание базовой конфигурации для работы с git

Установим git, зададим имя и почту владельца, настроим utf-8, зададим имя начальной ветки, autocrlf, safecrlf

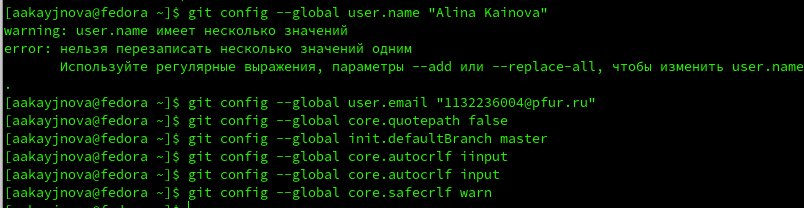


Figure 1: Базовая настройка git

## 4.2 Создание ключей SSH

Создадим ключ по алгоритму rsa размером 4096 бит

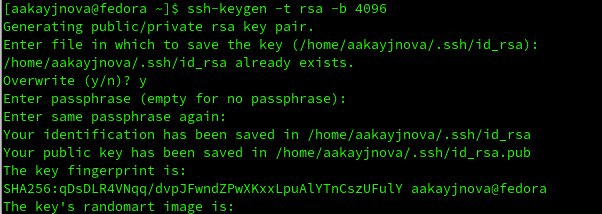


Figure 2: Ключ SSH по алгоритму rsa

Создадим ключ по алгоритму ed25519

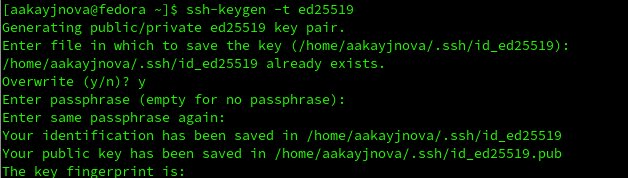


Figure 3: Ключ SSH по алгоритму ed25519

## 4.3 Создание ключа PGP

Генерируем ключ и выбираем заданные опции

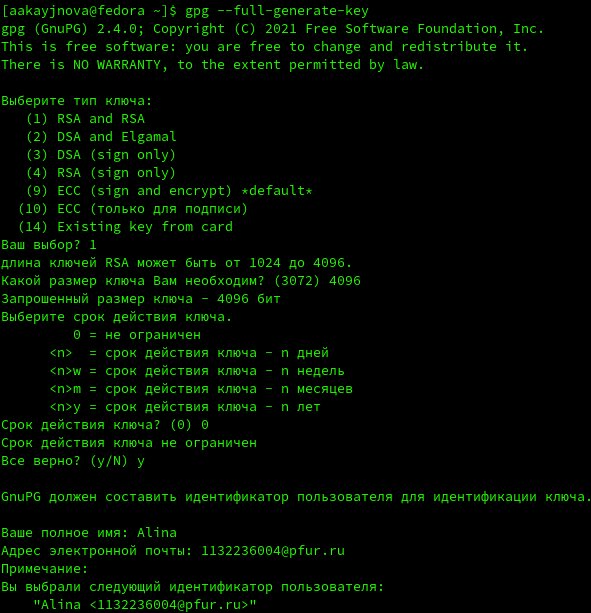


Figure 4: Ключ PGP

## 4.4 Настройка подписи git

Указываем Git применять введённый email при подписи коммитов

Figure 5: Настройка автоматических подписей коммитов git

Figure 5: Настройка автоматических подписей коммитов git

## 4.5 Регистрация на Github

У меня уже есть аккаунт на Github

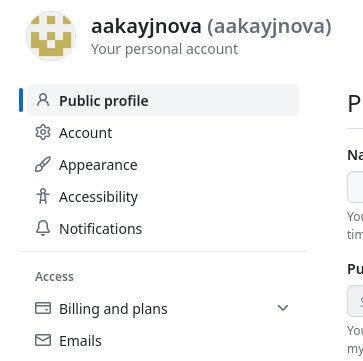


Figure 6: Аккаунт git

## 4.6 Создание локального каталога для выполнения заданий по предмету

Создадим шаблон рабочего пространства

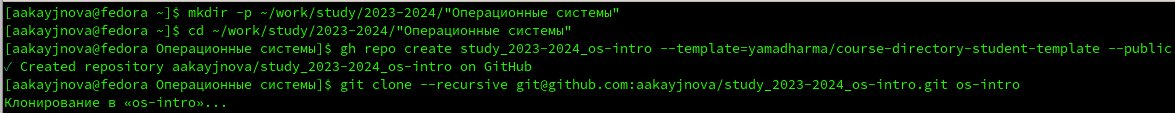


Figure 7: Создание репозитория курса на основе шаблона

Настроим каталог курса

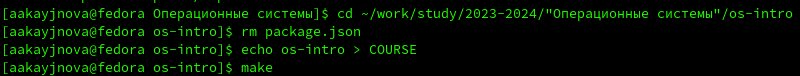


Figure 8: Настройка каталога курса

Отправим файлы на сервер

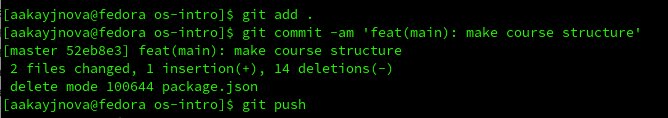


Figure 9: Отправка файлов

# 5 Выводы

В ходе данной лабораторной работы мы изучили идеологию и применение средств контроля версий, также приобрели навыки по работе с git.

# Список литературы

1. https://esystem.rudn.ru/mod/page/view.php?id=1098790#orgf425532

::: :::