Лабораторная работа №2

Отчёт

Александр Денисович Мосолов

Содержание

# 1 Цель работы

Изучить идеологию и применение средств контроля версий.  
Освоить умения по работе с git. [1]

# 2 Задание

Научиться применять команды git, работать с github.

# 3 Выполнение лабораторной работы

Установим git и gh (рис. 1):

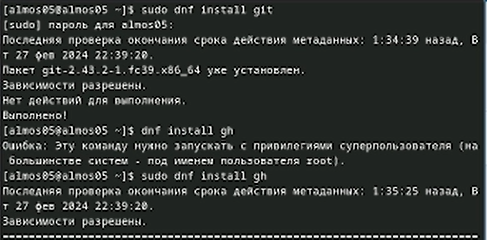


Рис. 1: Установка git и gh

Зададим имя и email владельца репозитория (рис. 2):  
*git config –global user.name “Aleksandr Mosolov”*  
*git config –global user.email “tenderboylive2@gmail.com”*:

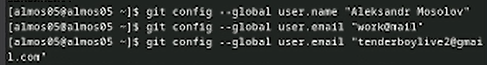


Рис. 2: Задаём имя и email

Настроим utf-8 в выводе сообщений git Зададим имя начальной ветки (будем называть её master) Параметр autocrlf Параметр safecrlf (рис. 3):

Настраиваем кодировку, задаём имя ветки и указываем параметры

Рис. 3: Настраиваем кодировку, задаём имя ветки и указываем параметры

По алгоритму rsa с ключём размером 4096 бит (рис. 4):  
*ssh-keygen -t rsa -b 4096*

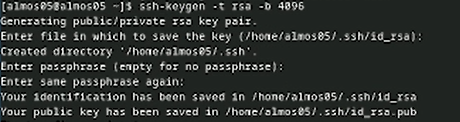


Рис. 4: Создаём ключ по алгоритму rsa

По алгоритму ed25519 (рис. 5):  
*ssh-keygen -t ed25519*

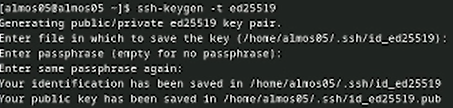


Рис. 5: Создаём ключ по алгоритму ed25519

Генерируем ключ (рис. 6):  
*gpg –full-generate-key*

Из предложенных опций выбираем:  
тип RSA and RSA; размер 4096; выберите срок действия; значение по умолчанию — 0 (срок действия не истекает никогда).

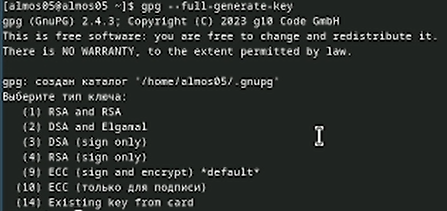


Рис. 6: Генерируем ключ

Выводим список ключей и копируем отпечаток приватного ключа:  
*gpg –list-secret-keys –keyid-format LONG*

Выводим и копируем ключ (рис. 7):  
*gpg –armor –export tenderboylive2@gmail.com | cat*



Рис. 7: Выводим и копируем ключ

Переходим в настройки GitHub (https://github.com/settings/keys), нажмаем на кнопку New GPG key и вставляем полученный ключ в поле ввода.

Настраиваем конфиг (рис. 8):  
*git config –global user.signingkey tenderboylive2@gmail.com*  
*git config –global commit.gpgsign true*  
*git config –global gpg.program $(which gpg2)*



Рис. 8: Настраиваем конфиг

Необходимо создать шаблон рабочего пространства (рис. 9):  
*mkdir -p ~/work/study/2023-2024/“Операционные системы”*  
*cd ~/work/study/2023-2024/“Операционные системы”*  
*gh repo create study\_2023-2024\_os-intro –template=yamadharma/course-directory-student-template –public*  
*git clone –recursive git@github.com:/study\_2023-2024\_os-intro.git os-intro*

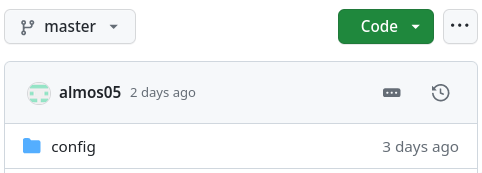


Рис. 9: Материалы курса на моём github

Переходим в каталог курса и удаляем лишние файлы (рис. 10):  
*cd ~/work/study/2023-2024/“Операционные системы”/os-intro*  
*rm package.json*

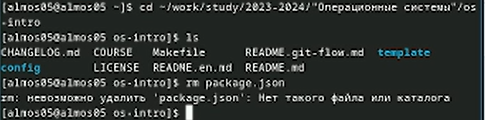


Рис. 10: Удаляем лишние файлы

Создаём необходимые каталоги (рис. 11):  
*echo os-intro > COURSE*  
*make*

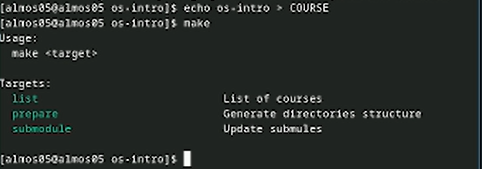


Рис. 11: Создаём необходимые каталоги

Фиксируем изменения на github (рис. 12):  
*git add .*  
*git commit -am ‘feat(main): make course structure’*  
*git push*

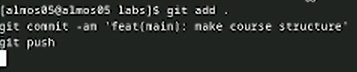


Рис. 12: Фиксируем изменения на github

# 4 Выводы

В ходе работы мы изучили идеологию и применение средств контроля версий. Освоили умения по работе с git.

# Список литературы

1. Kulyabov. [Архитектура компьютеров и операционные системы. Раздел "Операционные системы" (09.03.03, НПИбд)](https://esystem.rudn.ru/mod/page/view.php?id=1098933).