

# Отчёт по лабораторной работе

## Дисциплина: Операционные системы

Елена Михайловна Горпинич

### Содержание

#### Цель работы

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

#### Задание

Создать базовую конфигурацию для работы с git  
Создать ключ SSH  
Создать ключ PGP  
Настроить подписи git  
Зарегистрироваться на GitHub  
Создать локальный каталог для выполнения заданий по предмету

#### Теоретическое введение

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. ## Основные команды git  
`git add` Команда добавляет содержимое рабочего каталога в индекс для последующего коммита.  
`git status` Команда `git status` показывает состояния файлов в рабочем каталоге и индексе  
`git diff` Команда `git diff` используется для вычисления разницы между любыми двумя Git деревьями.

`git difftool` Команда `git difftool` просто запускает внешнюю утилиту сравнения для показа различий в двух деревьях, на случай если вы хотите использовать что-либо отличное от встроенного просмотрщика `git diff`.

`git commit` Команда `git commit` берёт все данные, добавленные в индекс с помощью `git add`, и сохраняет их слепок во внутренней базе данных, а затем сдвигает указатель текущей ветки на этот слепок.

`git reset` Команда используется в основном для отмены изменений.

`git rm` Команда `git rm` используется в Git для удаления файлов из индекса и рабочей копии.

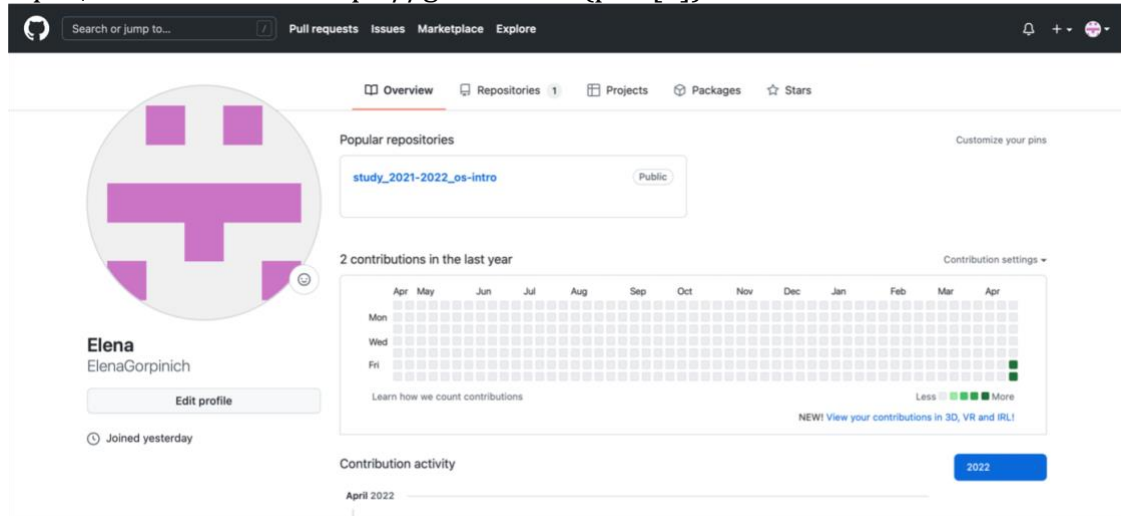
git mv Команда git mv — это всего лишь удобный способ переместить файл, а затем выполнить git add для нового файла и git rm для старого.

git clean Команда git clean используется для удаления мусора из рабочего каталога.

## Выполнение лабораторной работы

### Настройка GitHub

- 1) Изначально создаём учетную запись и заполняем основные данные на официальном сайте. <https://github.com>. (рис.[1])



- 2) Затем синхронизируем учетную запись с компьютером с помощью терминала. (рис.[2])

```
emgorpinich@emgorpinich:~$ git config --global user.name "Elena Gorpinich"
emgorpinich@emgorpinich:~$ git config --global user.email "gorpinich-lena@mail.ru"
```

- 3) Начинаем базовую настройку git: настроим utf-8 в выводе сообщений git, верификацию и подписание коммитов git, зададим имя начальной ветки (будем называть её master), параметр autocrlf, параметр safecrlf (рис.[3])

```
emgorpinich@emgorpinich:~$ git config --global core.quotePath false
emgorpinich@emgorpinich:~$ git config --global init.defaultBranch master
emgorpinich@emgorpinich:~$ git config --global core.autocrlf input
emgorpinich@emgorpinich:~$ git config --global core.safecrlf warn
```

- 4) Создадим ключ ssh по алгоритму rsa с ключевым размером 4096 бит (рис.[4])

```
emgorpinich@emgorpinich:~$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/emgorpinich/.ssh/id_rsa): lab2_1
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in lab2_1
Your public key has been saved in lab2_1.pub
The key fingerprint is:
SHA256:C4qjXaz9nBcwua33lXF1skiX1LSYb/BFk8bCHBKl1RY emgorpinich@emgorpinich
The key's randomart image is:
+---[RSA 4096]---+
|                 o+=E=|
|                 +=oO+|
|      .      ..+B.+|
|      +      . o++o|
|      .=S      o o+|
|      o ...o.   +.|
|      o +   ... o|
|      o = ...o .|
|      . o ..+o ..|
+-----[SHA256]-----+
```

- 5) Создадим ключ pgr. Генерируем ключ и из предложенных опций выбираем опции описанные в условии лабораторной работы. (рис.[5])

```
emgorpinich@emgorpinich:~$ gpg --full-generate-key
gpg (GnuPG) 2.2.19; Copyright (C) 2019 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.

Please select what kind of key you want:
  (1) RSA and RSA (default)
  (2) DSA and Elgamal
  (3) DSA (sign only)
  (4) RSA (sign only)
  (14) Existing key from card
Your selection? 1
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (3072) 4096
Requested keysize is 4096 bits
Please specify how long the key should be valid.
    0 = key does not expire
    <n> = key expires in n days
    <n>w = key expires in n weeks
    <n>m = key expires in n months
    <n>y = key expires in n years
Key is valid for? (0) 0
Key does not expire at all
Is this correct? (y/N) y

GnuPG needs to construct a user ID to identify your key.

Real name: ElenaGorpinich
Email address: gorpinich-lena@mail.ru
Comment:
You selected this USER-ID:
    "ElenaGorpinich <gorpinich-lena@mail.ru>"
```

- 6) Добавим PGP ключ в GitHub. Для этого выводим список ключей и скопируем отпечаток приватного ключа. Скопируем наш сгенерированный PGP ключ в буфер обмена. С помощью настроек GitHub (<https://github.com/settings/keys>),

добавим наш скопированный ключ, вставив его в поле ввода, которое появилось после нажатия кнопки New GPG key (рис.[7-9])

```
emgorpinich@emgorpinich:~$ gpg --list-secret-keys --keyid-format LONG
/home/emgorpinich/.gnupg/pubring.kbx
-----
sec   rsa4096/46A3B8E79662EF24 2022-04-23 [SC]
      75F4D4D859A5758BADDECA5E46A3B8E79662EF24
uid   [ultimate] ElenaGorpinich <gorpinich-lena@mail.ru>
ssb   rsa4096/A91F019269262B4D 2022-04-23 [E]

emgorpinich@emgorpinich:~$ gpg --armor --export 46A3B8E79662EF24 | xclip -sel clip
```

GPG keys / Add new

Key

Begins with '-----BEGIN PGP PUBLIC KEY BLOCK-----'

Add GPG key

GPG keys

New GPG key

This is a list of GPG keys associated with your account. Remove any keys that you do not recognize.



Email address: gorpinich-lena@mail.ru

Key ID: 46A3B8E79662EF24

Subkeys: A91F019269262B4D

Added on 23 Apr 2022

Delete

Learn how to [generate a GPG key and add it to your account](#).

- 7) Настроим автоматических подписей коммитов git (рис.[10])

```
emgorpinich@emgorpinich:~$ git config --global user.signingkey 46A3B8E79662EF24
emgorpinich@emgorpinich:~$ git config --global commit.gpgsign true
emgorpinich@emgorpinich:~$ git config --global gpg.program $(which gpg2)
```

- 8) Создадим новый репозиторий курса на основе шаблона и далее настроим каталог курса, удалив ненужные файлы, создав необходимые каталоги и отправив файлы на сервер (рис.[11-12])

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Owner \*



ElenaGorpinich ▾

Repository name \*

study\_2021-2022\_os-intro ✓

Great repository names are short and lowercase. Your new repository will be created as `study_2021-2022_os-intro-username`. [Learn more](#).

Description (optional)



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☐ Add a README file

This is where you can write a long description for your project. [Learn more](#).

Add .gitignore

Choose which files not to track from a list of templates. [Learn more](#).

.gitignore template: None ▾

Choose a license

A license tells others what they can and can't do with your code. [Learn more](#).

License: None ▾

You are creating a public repository in your personal account.

Create repository



Elena

ElenaGorpinich

Edit profile

Overview Repositories 1 Projects Packages Stars

Find a repository...

Type ▾

Language ▾

Sort ▾

New

study\_2021-2022\_os-intro Public

Updated 1 hour ago

Star ▾

Вывод

#

В данной лабораторной работе я научилась работать с Github (создавать и привязывать учетную запись к компьютеру). Разобрала основные команды git и рассмотрела, как их применять их при работе с Github. Изучила идеологию и научилась применять средства контроля версий

## Список литературы