

Первоначальна настройка git

Лабораторная работа 2

Провоторов Антон

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выводы	13

Список иллюстраций

4.1	Задача параметров git	8
4.2	Создание ssh ключа 1	8
4.3	Создание ssh ключа 2	9
4.4	Создание gpg ключа 1	9
4.5	Создание gpg ключа 2	10
4.6	Список ключей	10
4.7	Скопируем ключ	10
4.8	Ключ на github	10
4.9	Настройка автоматических подписей коммитов git	11
4.10	Авторизация gh	11
4.11	Настройка рабочего пространства	11
4.12	Изменение данных	11
4.13	git add, git commit	12
4.14	git push	12

Список таблиц

1 Цель работы

Освоение навыков работы с git

2 Задание

Настроить git на ПК, и выполнить действия с ним

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.

4 Выполнение лабораторной работы

Укажем данные владельца git и зададим параметры(рис. 4.1).

```
agprovotorov@dk5n51 ~ $ git config --global user.name "AntonProvotorov"
agprovotorov@dk5n51 ~ $ git config --global user.email "1132230300@pfur.ru"
agprovotorov@dk5n51 ~ $ git config --global core.quotepath false
agprovotorov@dk5n51 ~ $ git config --global init.defaultBranch master
agprovotorov@dk5n51 ~ $ git config --global core.autocrlf input
agprovotorov@dk5n51 ~ $ git config --global core.safecrlf warn
```

Рис. 4.1: Задача параметров git

Создадим ssh ключ по алгоритму rsa размером 4096 бит(рис. 4.2).

```
agprovotorov@dk5n51 ~ $ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/afs/.dk.sci.pfu.edu.ru/home/a/g/agprovotorov/.ssh/id_rsa):
/afs/.dk.sci.pfu.edu.ru/home/a/g/agprovotorov/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /afs/.dk.sci.pfu.edu.ru/home/a/g/agprovotorov/.ssh/id_rsa
Your public key has been saved in /afs/.dk.sci.pfu.edu.ru/home/a/g/agprovotorov/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:IV1jKZ1JJYUt6yRd0R91R1JysyVJrUB0+sB8y04QHvQ agprovotorov@dk5n51
The key's randomart image is:
+---[RSA 4096]---+
|          o*%X+*X|
|       ..oB*+*oB*|
|      .oo +B.E+ .|
|     ...+ *...   |
|      S+  =      |
|      .  o       |
|      .           |
|      .           |
+---[SHA256]-----+
```

Рис. 4.2: Создание ssh ключа 1

Теперь по алгоритму по алгоритму ed25519(рис. 4.3).


```

agprovotorov@dk5n51 ~ $ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/afs/.dk.sci.pfu.edu.ru/home/a/g/agprovotorov/.ssh/id_ed25519):
/afs/.dk.sci.pfu.edu.ru/home/a/g/agprovotorov/.ssh/id_ed25519 already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /afs/.dk.sci.pfu.edu.ru/home/a/g/agprovotorov/.ssh/id_ed25519.
Your public key has been saved in /afs/.dk.sci.pfu.edu.ru/home/a/g/agprovotorov/.ssh/id_ed25519.p
The key fingerprint is:
SHA256:fqvBSud8nfBvf5hgTDvpoeFvsWQIZNZFkrscK1wMUT0 agprovotorov@dk5n51
The key's randomart image is:
+--[ED25519 256]--+
|      .+0+      |
|      + o.o E   |
|      + . + .   |
|      . * .     |
|      So B o    |
|      o * %     |
|      . = . @ B o |
|      . = o+.B + . |
|      . +o.o.o..o |
| +-----[SHA256]-----+

```

Рис. 4.3: Создание ssh ключа 2

Создадим gpg ключ(рис. 4.4).

```

agprovotorov@dk5n51 ~ $ gpg --full-generate-key
gpg (GnuPG) 2.2.42; Copyright (C) 2023 g10 Code GmbH
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Выберите тип ключа:
  (1) RSA и RSA (по умолчанию)
  (2) DSA и Elgamal
  (3) DSA (только для подписи)
  (4) RSA (только для подписи)
  (14) Имеющийся на карте ключ
Ваш выбор? 1
длина ключей RSA может быть от 1024 до 4096.
Какой размер ключа Вам необходим? (3072) 4096
Запрошенный размер ключа - 4096 бит
Выберите срок действия ключа.
  0 = не ограничен
  <n> = срок действия ключа - n дней
  <n>w = срок действия ключа - n недель
  <n>m = срок действия ключа - n месяцев
  <n>y = срок действия ключа - n лет
Срок действия ключа? (0)
Срок действия ключа не ограничен
Все верно? (y/N) y

GnuPG должен составить идентификатор пользователя для идентификации ключа.

Ваше полное имя: agprovotorov
Адрес электронной почты: 1132230300@pfur.ru
Примечание:
Вы выбрали следующий идентификатор пользователя:
  "agprovotorov <1132230300@pfur.ru>"

```

Рис. 4.4: Создание gpg ключа 1

(к сожалению, на один скриншот не уместился весь процесс) (рис. 4.5).

```
длина ключей RSA может быть от 1024 до 4096.
Какой размер ключа Вам необходим? (3072) 4096
Запрошенный размер ключа - 4096 бит
Выберите срок действия ключа.
0 = не ограничен
<п> = срок действия ключа - п дней
<пж> = срок действия ключа - п недель
<пм> = срок действия ключа - п месяцев
<пл> = срок действия ключа - п лет
Срок действия ключа? (0)
Срок действия ключа не ограничен
Все верно? (y/N) y

GnuPG должен составить идентификатор пользователя для идентификации ключа.

Ваше полное имя: agprovotorov
Адрес электронной почты: 1132230300@pfur.ru
Примечание:
Вы выбрали следующий идентификатор пользователя:
"agprovotorov <1132230300@pfur.ru>"

Сменить (N)Имя, (C)Примечание, (E)Адрес; (O)Принять/(Q)Выход?
Сменить (N)Имя, (C)Примечание, (E)Адрес; (O)Принять/(Q)Выход? O
Необходимо получить много случайных чисел. Хелательно, чтобы Вы
в процессе генерации выполняли какие-то другие действия (печать
на клавиатуре, движение мыши, обращение к диску); это даст генератору
случайных чисел больше возможностей получить достаточное количество энтропии.
Необходимо получить много случайных чисел. Хелательно, чтобы Вы
в процессе генерации выполняли какие-то другие действия (печать
на клавиатуре, движение мыши, обращение к диску); это даст генератору
случайных чисел больше возможностей получить достаточное количество энтропии.
gpg: сертификат отныне записан в '/afs/.dk.sci.pfu.edu.ru/home/a/g/agprovotorov/.gnupg/openpgp-revocs.d/24013101604D9E56699ECE183F4B8CFB5ABC49AB.rev'.
Открытый и секретный ключи созданы и подписаны.

pub rsa4096 2024-02-29 [SC]
24013101604D9E56699ECE183F4B8CFB5ABC49AB
uid agprovotorov <1132230300@pfur.ru>
sub rsa4096 2024-02-29 [E]
```

Рис. 4.5: Создание gpg ключа 2

Выведем список gpg ключей (рис. 4.6).

```
agprovotorov@dk5n51 ~ $ gpg --list-secret-keys --keyid-format LONG
gpg: проверка таблицы доверия
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: глубина: 0 достоверных: 2 подписанных: 0 доверие: 0~, 0q, 0n, 0f, 2u
/afs/.dk.sci.pfu.edu.ru/home/a/g/agprovotorov/.gnupg/pubring.kbx
-----
sec rsa4096/0C1A34D33EEC2113 2024-02-29 [SC]
A569FF6153CB11929156F6730C1A34D33EEC2113
uid [ абсолютно ] agprovotorov <1132230300@pfur.ru>
ssb rsa4096/3ECDF34A99BBB728 2024-02-29 [E]

sec rsa4096/3F4B8CFB5ABC49AB 2024-02-29 [SC]
24013101604D9E56699ECE183F4B8CFB5ABC49AB
uid [ абсолютно ] agprovotorov <1132230300@pfur.ru>
ssb rsa4096/736C0161B8D1942C 2024-02-29 [E]
```

Рис. 4.6: Список ключей

Скопируем ключ gpg в буфер обмена (рис. 4.7).

```
agprovotorov@dk5n51 ~ $ gpg --armor --export 3F4B8CFB5ABC49AB | xclip -sel clip
```

Рис. 4.7: Скопируем ключ

Загрузим его на свой github (рис. 4.8).

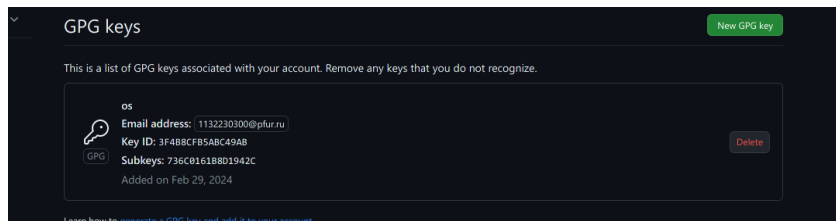


Рис. 4.8: Ключ на github

Сделаем настройку автоматических подписей коммитов git(рис. 4.9).

```
agprovotorov@dk5n51 ~ $ gpg --armor --export 3F4B8CFB5ABC49AB | xclip -sel clip
agprovotorov@dk5n51 ~ $ git config --global user.signingkey 3F4B8CFB5ABC49AB
agprovotorov@dk5n51 ~ $ git config --global commit.gpgsign true
agprovotorov@dk5n51 ~ $ git config --global gpg.program $(which gpg2)
```

Рис. 4.9: Настройка автоматических подписей коммитов git

Авторизуемся в gh (рис. 4.10).

```
agprovotorov@dk5n51 ~ $ gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations on this host? SSH
? Upload your SSH public key to your GitHub account? /afs/.dk.sci.pfu.edu.ru/home/a/g/agprovotorov/.ssh/id_rsa.pub
? Title for your SSH key: title
? How would you like to authenticate GitHub CLI? Login with a web browser
! First copy your one-time code: 4AB8-1F6B
Press Enter to open github.com in your browser...
Authentication complete.
$ gh config set -h github.com git_protocol ssh
Configured git protocol
! Uploaded the SSH key to your GitHub account: /afs/.dk.sci.pfu.edu.ru/home/a/g/agprovotorov/.ssh/id_rsa.pub
Logged in as AntonProvotorov
```

Рис. 4.10: Авторизация gh

Настроим рабочее пространство под себя и клонируем репозиторий (рис. 4.11).

```
agprovotorov@dk5n51 ~ $ mkdir -p ~/work/study/2022-2023/Операционные системы
agprovotorov@dk5n51 ~ $ cd ~/work/study/2022-2023/Операционные системы
agprovotorov@dk5n51 ~/work/study/2022-2023/Операционные системы $ gh repo create study_2022-2023-os-intro --template=yamadharma/course-directory-student-template --public
Created repository AntonProvotorov/study_2022-2023-os-intro on GitHub
agprovotorov@dk5n51 ~/work/study/2022-2023/Операционные системы $ cd
agprovotorov@dk5n51 ~ $ cd ~/work/study/2023-2024/Операционные системы
agprovotorov@dk5n51 ~/work/study/2023-2024/Операционные системы $ git clone --recursive git@github.com:owner/study_2023-2024-os-intro.git os-intro
bash: owner: Нет такого файла или каталога
agprovotorov@dk5n51 ~/work/study/2023-2024/Операционные системы $ git clone --recursive git@github.com:owner/study_2022-2023-os-intro.git os-intro
bash: owner: Нет такого файла или каталога
agprovotorov@dk5n51 ~/work/study/2023-2024/Операционные системы $ git clone --recursive git@github.com:owner/study_2023-2024-os-intro.git os-intro
bash: owner: Нет такого файла или каталога
agprovotorov@dk5n51 ~/work/study/2023-2024/Операционные системы $ git clone --recursive git@github.com:AntonProvotorov/study_2022-2023-os-intro.git os-intro
Клонирование в «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
Получение объектов: 100% (32/32), 16.68 KiB | 952.00 KiB/c, готово.
Определение изменений: 100% (1/1), готово.
Updating files: 100% (28/28), готово.
Подмодуль «template/presentation» (https://github.com/yamadharma/academic-presentation-markdown-template.git) зарегистрирован по пути «template/presentation»
Подмодуль «template/report» (https://github.com/yamadharma/academic-laboratory-report-template.git) зарегистрирован по пути «template/report»
Клонирование в «/afs/.dk.sci.pfu.edu.ru/home/a/g/agprovotorov/work/study/2023-2024/Операционные системы/os-intro/template/presentation»...
remote: Enumerating objects: 95, done.
remote: Counting objects: 100% (95/95), done.
remote: Compressing objects: 100% (67/67), done.
```

Рис. 4.11: Настройка рабочего пространства

Теперь изменим данные в нашем скопированном репозитории (рис. 4.12).

```
agprovotorov@dk5n51 ~/work/study/2023-2024/Операционные системы $ cd ~/work/study/2023-2024/Операционные системы/os-intro
agprovotorov@dk5n51 ~/work/study/2023-2024/Операционные системы/os-intro $ rm package.json
agprovotorov@dk5n51 ~/work/study/2023-2024/Операционные системы/os-intro $ echo os-intro > COURSE
agprovotorov@dk5n51 ~/work/study/2023-2024/Операционные системы/os-intro $ make
Usage:
  make <target>

Targets:
  list      List of courses
  prepare   Generate directories structure
  submodule Update submodules
```

Рис. 4.12: Изменение данных

Выполним команду git add, git commit (рис. 4.13).

```

agprovotorov@dk5n51 ~/work/study/2023-2024/Операционные системы/os-intro $ git add .
agprovotorov@dk5n51 ~/work/study/2023-2024/Операционные системы/os-intro $ git commit -am 'feat(main): make course structure'
[main f6b6e12] feat(main): make course structure
2 files changed, 1 insertion(+), 14 deletions(-)
delete mode 100644 package.json

```

Рис. 4.13: git add, git commit

В завершение выполним git push, и завершим работу(рис. 4.14).

```

agprovotorov@dk5n51 ~/work/study/2023-2024/Операционные системы/os-intro $ git push
Перечисление объектов: 5, готово.
Подсчет объектов: 100% (5/5), готово.
При скачивании изменений используется до 6 потоков
Сжатие объектов: 100% (2/2), готово.
Запись объектов: 100% (3/3), 952 байта | 952.00 КБ/с, готово.
Всего 3 (изменений 1), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote: This repository moved. Please use the new location:
remote:   git@github.com:AntonProvotorov/study_2023-2024-os-intro.git
To github.com:AntonProvotorov/study_2023-2024-os-intro.git
   b50b583..f6b6e12 master -> master

```

Рис. 4.14: git push

5 Выводы

Мы научились базовой работе с гит