

Лабораторная работа №4

Презентация

Мосолов А.Д.

14 марта 2024

Российский университет дружбы народов, Москва, Россия

- Мосолов Александр Денисович
- Студент, НПИбд0?-23
- Российский университет дружбы народов
- 1132236128@pfur.ru

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

Научиться применять команды git, работать с github.

Установка git и gh

Установим git и gh:

```
[almos05@almos05 ~]$ sudo dnf install git
[sudo] пароль для almos05:
Последняя проверка окончания срока действия метаданных: 1:34:39 назад, в
т 27 фев 2024 22:39:20.
Пакет git-2.43.2-1.fc39.x86_64 уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
[almos05@almos05 ~]$ dnf install gh
Ошибка: Эту команду нужно запускать с привилегиями суперпользователя (на
большинстве систем - под именем пользователя root).
[almos05@almos05 ~]$ sudo dnf install gh
Последняя проверка окончания срока действия метаданных: 1:35:25 назад, в
т 27 фев 2024 22:39:20.
Зависимости разрешены.
```

Рис. 1: Установка git и gh

Задаём имя и email

Зададим имя и email владельца репозитория:

git config --global user.name "Aleksandr Mosolov"

git config --global user.email "tenderboylive2@gmail.com":

```
[almos05@almos05 ~]$ git config --global user.name "Aleksandr Mosolov"  
[almos05@almos05 ~]$ git config --global user.email "work@mail"  
[almos05@almos05 ~]$ git config --global user.email "tenderboylive2@gmail  
l.com"
```

Рис. 2: Задаём имя и email

Настраиваем кодировку, задаём имя ветки и указываем параметры

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

```
[almos05@almos05 ~]$ git config --global core.quotepath false  
[almos05@almos05 ~]$ git config --global init.defaultBranch master  
[almos05@almos05 ~]$ git config --global core.autocrlf input  
[almos05@almos05 ~]$ git config --global core.safecrlf warn
```

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

Создаём ключ по алгоритму rsa

По алгоритму rsa с ключём размером 4096 бит:

ssh-keygen -t rsa -b 4096

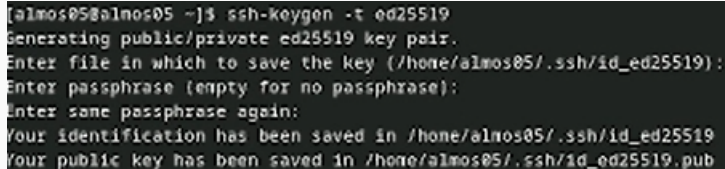
```
[almos05@almos05 ~]$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/almos05/.ssh/id_rsa):
Created directory '/home/almos05/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/almos05/.ssh/id_rsa
Your public key has been saved in /home/almos05/.ssh/id_rsa.pub
```

Рис. 4: Создаём ключ по алгоритму rsa

Создаём ключ по алгоритму ed25519

По алгоритму ed25519:

ssh-keygen -t ed25519

A terminal window showing the command 'ssh-keygen -t ed25519' being executed. The output shows the generation of a public/private key pair, prompts for a file name, passphrase, and confirmation of the passphrase, and finally the saving of the key files.

```
[almos05@almos05 ~]$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/almos05/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/almos05/.ssh/id_ed25519
Your public key has been saved in /home/almos05/.ssh/id_ed25519.pub
```

Рис. 5: Создаём ключ по алгоритму ed25519

Генерируем ключ

Генерируем ключ:

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).

```
[almos05@almos05 ~]$ gpg --full-generate-key
gpg (GnuPG) 2.4.3; 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.

gpg: создан каталог '/home/almos05/.gnupg'
Выберите тип ключа:
  (1) RSA and RSA
  (2) DSA and ElGamal
  (3) DSA (sign only)
  (4) RSA (sign only)
  (9) ECC (sign and encrypt) *default*
 (10) ECC (только для подписи)
 (14) Existing key from card
```

Выводим и копируем ключ

Выводим список ключей и копируем отпечаток приватного ключа:

```
gpg --list-secret-keys --keyid-format LONG
```

Выводим и копируем ключ:

```
gpg --armor --export tenderboylive2@gmail.com | cat
```

```
(almos05@almos05 ~)$ gpg --armor --export tenderboylive2@gmail.com | cat  
-----BEGIN PGP PUBLIC KEY BLOCK-----
```

```
MIIB3GXEUnSBLADKRncqkRIIMacIH6G1n6eHYxbFuLnt3PgW7DSwhc/yqJ5dw0uJH  
98gfsy8oSHXpywv8Xhzoy0DF3X0w7VpARACr0KQBjRcg3iWLy3MdNQhyXvho99/I  
QLyn53BXaGPfTcs08q5D11W=810wbEdWGLE71jTg9nWaoHzylzJjggSf553cRx5C  
GRJHaa919W6ac5D51K8uA614XW1CCTB67GBzN13ARZT1PS101f-Chd5wfbPbgwQ2  
KiKdHz0x/yHbjJCwgZS/nmeX21nzuRGIj7GYZZDK9BjJT2p3sU3SUPHJ792RwYou  
gpnHzIF0f6Gk57f2qXvPIIn5drzqBkp38SSqA5nMtVAthGKHGVtwkTqxePbnZq+2v  
2qzQ+pei2sNeL6Up0RP4NJoVtsqg/cNDDtYgfeI2znKPDIZdHAQo31zNn5bn1R1r  
9Dgd5Ezb1U0aydjwd0g0k1f0rdhGzRL9MN+Yam4e0IAuLRfGt4H+V1xJFxAJcSU  
nzhX8dqZrwQ0qcE+FTuQFpnUDRapDhIKJd/6kYZoFon2R1DXVeUHRnEkq97nQ81N  
eWFebTHnGSL0F7vJtz1AGAULgXpwFcqsglZdXycKL0Sr3wgs8ALznISbPG+GvP3s  
nX9+jcfQME3B1mnd9-HzSjgLYN0dmHfLuVs1GRzN5kk2/ECuEUq08xoljQARAQAB  
tCxBBoGVrc2FuZHIgTM9zb2xvd1A8dGVuZGVyYn95bG12ZTJAZ21haWwY29tPokC  
UODT6Dn6DnXbD3+GonUN5cEAGEU1UeTR06uDS92H0L131L7AbcVMDcJCAcCA1Jf
```

Настраиваем конфиг

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

Настраиваем конфиг:

```
git config --global user.signingkey tenderboylive2@gmail.com
```

```
git config --global commit.gpgsign true
```

```
git config --global gpg.program $(which gpg2)
```

```
[almos05@almos05 ~]$ git config --global user.signingkey tenderboylive2@gmail.com
[almos05@almos05 ~]$ git config --global commit.gpgsign true
[almos05@almos05 ~]$ git config --global gpg.program $(which gpg2)
[almos05@almos05 ~]$ 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? /home/almos05/.ssh/id_rsa.pub
? Title for your SSH key: sway
? How would you like to authenticate GitHub CLI? [Use arrows to move, type to filter]
```

Материалы курса на моём github

Необходимо создать шаблон рабочего пространства:

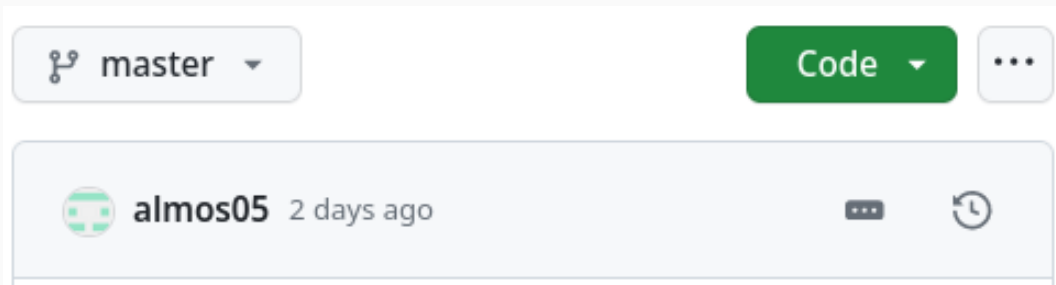
```
mkdir -p ~/work/study/2023-2024/“Операционные системы”
```

```
cd ~/work/study/2023-2024/“Операционные системы”
```

```
gh repo create study_2023-2024_os-intro
```

```
-template=yamadharma/course-directory-student-template --public
```

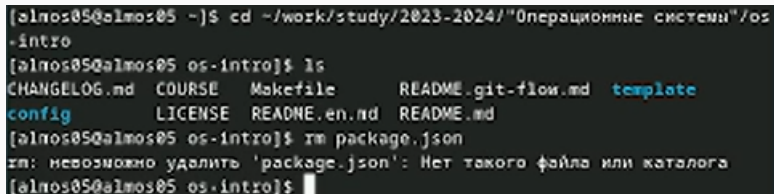
```
git clone --recursive git@github.com:/study_2023-2024_os-intro.git os-intro
```



Удаляем лишние файлы

Переходим в каталог курса и удаляем лишние файлы:

```
cd ~/work/study/2023-2024/"Операционные системы"/os-intro  
rm package.json
```

A terminal window with a dark background and light-colored text. The user navigates to the directory ~/work/study/2023-2024/"Операционные системы"/os-intro and lists the files. The files listed are CHANGELOG.md, COURSE, Makefile, README.git-flow.md, template, config, LICENSE, README.en.md, and README.md. The user then attempts to remove package.json, but the terminal shows an error message: "rm: невозможно удалить 'package.json': Нет такого файла или каталога".

```
[almos05@almos05 ~]$ cd ~/work/study/2023-2024/"Операционные системы"/os-intro  
[almos05@almos05 os-intro]$ ls  
CHANGELOG.md  COURSE  Makefile  README.git-flow.md  template  
config        LICENSE  README.en.md  README.md  
[almos05@almos05 os-intro]$ rm package.json  
rm: невозможно удалить 'package.json': Нет такого файла или каталога  
[almos05@almos05 os-intro]$
```

Рис. 10: Удаляем лишние файлы

Создаём необходимые каталоги

Создаём необходимые каталоги:

echo os-intro > COURSE

make



```
[almos05@almos05 os-intro]$ echo os-intro > COURSE
[almos05@almos05 os-intro]$ make
Usage:
  make <target>

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

[almos05@almos05 os-intro]$
```

Рис. 11: Создаём необходимые каталоги

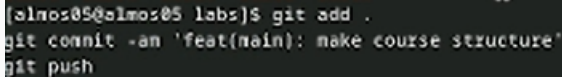
Фиксируем изменения на github

Фиксируем изменения на github:

git add .

git commit -am 'feat(main): make course structure'

git push

A screenshot of a terminal window with a black background and white text. It shows three lines of commands being entered: 'git add .', 'git commit -am 'feat(main): make course structure'', and 'git push'. The prompt '[almos05@almos05 labs]\$' is visible at the start of the first line.

```
[almos05@almos05 labs]$ git add .  
git commit -am 'feat(main): make course structure'  
git push
```

Рис. 12: Фиксируем изменения на github

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