

Лабораторная работа №2

Кривобоков Юрий Дмитриевич

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выводы	11
	Список литературы	12

Список иллюстраций

4.1	Конфиг	8
4.2	ключ	8
4.3	ключ в гх	9
4.4	гх авторизация	9
4.5	директория	9
4.6	мейк директори	10

Список таблиц

1 Цель работы

Целью данной работы является изучение идеологии и применения средств контроля версий и освоение умения по работе с git.

2 Задание

- базовую конфигурацию для работы с git.
- ключ SSH.
- ключ PGP.
- подписи git.
- на Github.
- 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. 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.

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 (пользователь) 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.

4 Выполнение лабораторной работы

Настроил конфиги (рис. -fig:001)

```
ydkrivobokov@vbox:~$ git config --global user.name "ydkrivobokov"
ydkrivobokov@vbox:~$ git config --global user.mail "urakrivobokov123@mail.ru"
ydkrivobokov@vbox:~$ git config --global core.quotepath false
ydkrivobokov@vbox:~$ git config --global init.defaultBranch master
ydkrivobokov@vbox:~$ git config --global core.autocrlf input
ydkrivobokov@vbox:~$ git config --global core.savecrlf warn
ydkrivobokov@vbox:~$
```

Рисунок 4.1: Конфиг

Сгенерировал ключ (рис. -fig:002)

```
ydkrivobokov@vbox:~$ git config --global core.autocrlf input
ydkrivobokov@vbox:~$ git config --global core.savecrlf warn
ydkrivobokov@vbox:~$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ydkrivobokov/.ssh/id_rsa):
Created directory '/home/ydkrivobokov/.ssh'.
Enter passphrase for "/home/ydkrivobokov/.ssh/id_rsa" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ydkrivobokov/.ssh/id_rsa
Your public key has been saved in /home/ydkrivobokov/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:q8WjyUv0fJGjc5lPjJV5NzC3p4lmjq1b4nbUM7G/Y6c ydkrivobokov@vbox
The key's randomart image is:
+---[RSA 4096]---+
|  o .               |
| . o + .           |
| o = 0 . o =       |
| = B S o +.oo      |
| . + o * . =       |
| o   +.o E         |
+---+-----+

```

Рисунок 4.2: ключ

Ввел публичный ключ в гитхаб (рис. -fig:003)


```
ydkrivobokov@vbox:~$ gpg --list-secret-keys --keyid-format LONG
[keybox]
-----
sec rsa4096/E6945721986FB700 2025-08-05 [SC]
      A240BC6F0AE7D038D9193A26E8945721986FB700
uid          [ unknown ] ydkrivobokov <ydkrivobokov123@mail.ru>
ssb rsa4096/E6945721986FB700 2025-08-05 [S]

ydkrivobokov@vbox:~$ gpg --armor --export E6945721986FB700
-----BEGIN PGP PUBLIC KEY BLOCK-----

mDIMEBESSE48E4Dp2pR0t6G6sWetg05NEEd'guj5L0qkMY3/Mic5JatY0UQR
In3YBISn1SekQK00Hkgal0a2jKqRCLFoR6vbaq?+wn2S8EgePkw4jnsjg3JAAG6
C3wvz05+FdEDc04t5EdXpGVBRJhE8og5I4B*N3EjDeV33VreTewr0ICrcFUR
C8Xj8+7X33W3q0f0UVTf0P04ESlgn7Ny7a0ead7HmH14Q1V0cpM2VP0ISY81V
TrQg2T0p;00eESER8sKa0q8o090yUfCLC'wefcJp0d70K50k1Lp6/31PH0W
MuyE0870+T07j+4+EcPol3C2y0L1r7a54jKc30oc+02waTny+0300/3FFn5000Q7
0Pft+M2xk10ENFKj0D720CH0cJ0eV0a023E+0v0u0gP7Vln0cM0H04w70u0WE
T53A+20031FAyY0NK/SU14qK17V03pww708ZFN/Z0d1weK1eeL030MX0E7S
```

Рисунок 4.3: ключ в гх

Авторизация через gh (рис. -fig:004)

```
ydkrivobokov@vbox:~$ jORUmWcI
=VvWc
-----END PGP PUBLIC KEY BLOCK-----
ydkrivobokov@vbox:~$ git config --global user.signingkey E6945721986FB700
ydkrivobokov@vbox:~$ git config --global commit.gpgsign true
ydkrivobokov@vbox:~$ git config --global gpg.program $(which gpg2)
ydkrivobokov@vbox:~$ gh auth login
? Where do you use GitHub? GitHub.com
? What is your preferred protocol for Git operations on this host? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Login with a web browser

! First copy your one-time code: 8BE4-9FFA
Press Enter to open https://github.com/login/device in your browser...
restorecon: SELinux: Could not get canonical path for /home/ydkrivobokov/.mozilla/firefox/
/gmp-widevinecdm/* restorecon: No such file or directory.
✓ Authentication complete.
- gh config set -h github.com git.protocol https
✓ Configured git protocol
✓ Logged in as ydkrivobokov
ydkrivobokov@vbox:~$
```

Рисунок 4.4: гх авторизация

Создал директорию (рис. -fig:005)

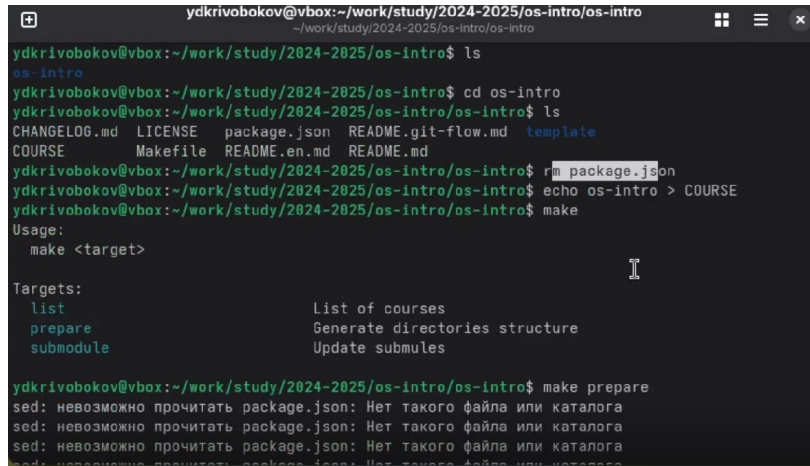
```
ydkrivobokov@vbox:~$ git config --global git.protocol https
✓ Configured git protocol
✓ Logged in as ydkrivobokov
ydkrivobokov@vbox:~$ cd
ydkrivobokov@vbox:~$ mkdir -p ~/work/study/2024-2025/os-intro
ydkrivobokov@vbox:~$ cd ~/work/study/2024-2025/os-intro
ydkrivobokov@vbox:~/work/study/2024-2025/os-intro$ gh repo create study_2024-2025_os-intro
--template+yamadharma/course-directory-student-template --public
unknown flag: --template+yamadharma/course-directory-student-template

Usage: gh repo create [<name>] [flags]

Flags:
  --add-readme      Add a README file to the new repository
```

Рисунок 4.5: директория

Установил ос-интро и отправил на сервер (рис. -fig:006)



```
ydkrivobokov@vbox:~/work/study/2024-2025/os-intro/os-intro$ ls
os-intro
ydkrivobokov@vbox:~/work/study/2024-2025/os-intro$ cd os-intro
ydkrivobokov@vbox:~/work/study/2024-2025/os-intro/os-intro$ ls
CHANGELOG.md LICENSE package.json README.git-flow.md template
COURSE Makefile README.en.md README.md
ydkrivobokov@vbox:~/work/study/2024-2025/os-intro/os-intro$ rm package.json
ydkrivobokov@vbox:~/work/study/2024-2025/os-intro/os-intro$ echo os-intro > COURSE
ydkrivobokov@vbox:~/work/study/2024-2025/os-intro/os-intro$ make
Usage:
  make <target>

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

ydkrivobokov@vbox:~/work/study/2024-2025/os-intro/os-intro$ make prepare
sed: невозможно прочитать package.json: Нет такого файла или каталога
sed: невозможно прочитать package.json: Нет такого файла или каталога
sed: невозможно прочитать package.json: Нет такого файла или каталога
sed: невозможно прочитать package.json: Нет такого файла или каталога
```

Рисунок 4.6: мейк директори

5 Выводы

В результате выполнения данной лабораторной работы я приобрел необходимые навыки работы с гит, научился созданию репозитория, gpg и ssh ключей, настроил каталог курса и авторизовался в gh.

Список литературы