

Отчёт по лабораторной работе № 2

Первоначальна настройка git.

Сергей Витальевич Павлюченков

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	17
5	Контрольные вопросы	18

Список иллюстраций

3.1	Установка git	7
3.2	Установка gh	7
3.3	Использование git config	8
3.4	Создание SSH-ключа	9
3.5	Создание PGP-ключа	10
3.6	Полученные PGP-ключи	10
3.7	Настройки github	11
3.8	Заполнение ключа	12
3.9	Новый ключ	13
3.10	Настраиваю подписи коммитов git.	13
3.11	Авторизация gh	14
3.12	Создание репозитория курса на основе шаблона	15
3.13	Настройка каталога курса	15
3.14	Создание необходимых каталогов	16
3.15	Использование	16

Список таблиц

1 Цель работы

Изучить идеологию и применение средств контроля версий.

Освоить умения по работе с git.

2 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.

3 Выполнение лабораторной работы

Устанавливаю git командой `dnf install git`.

```
[svpavluchenkov@fedora ~]$ sudo -i
[sudo] пароль для svpavluchenkov:
[root@fedora ~]# dnf -y install git
Последняя проверка окончания срока действия метаданных: 2:24:39 назад, Пт 21 июн 2024 18:33:59.
Пакет git-2.45.2-2.fc40.x86_64 уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
[root@fedora ~]#
```

Рис. 3.1: Установка git

Устанавливаю gh командой `dnf install gh`.

```
[root@fedora ~]# dnf -y install gh
Последняя проверка окончания срока действия метаданных: 2:25:27 назад, Пт 21 июн 2024 18:33:59.
Зависимости разрешены.
=====
Пакет                Архитектура          Версия
=====
Установка:
gh                    x86_64                2.45.0-1.fc40
=====
Результат транзакции
=====
Установка 1 Пакет
=====
Объем загрузки: 8.7 М
Объем изменений: 43 М
Загрузка пакетов:
gh-2.45.0-1.fc40.x86_64.rpm
=====
Общий размер
Проверка транзакции
Проверка транзакции успешно завершена.
Идет проверка транзакции
```

Рис. 3.2: Установка gh

Задаю имя и email владельца репозитория, настраиваю utf-8 в выводе сообщений git, задаю имя начальной ветки и меняю параметры `autocrlf` и `safecrlf`. (рис. 3.1).

```
root@fedora ~]# git config --global user.name "Sergey Pavliuchenkov"
root@fedora ~]# git config --global user.email "1132237372@pfur.ru"
root@fedora ~]# git config --global core.quotepath false
root@fedora ~]# git config --global init.defaultBranch master
root@fedora ~]# git config --global core.autocrlf input
root@fedora ~]# git config --global core.safecrlf warn
root@fedora ~]#
```

Рис. 3.3: Использование git config

Создаю ключи ssh по двум алгоритмам - по алгоритму rsa с ключём размером 4096 бит и по алгоритму ed25519(рис. 3.4).


```

[root@fedora ~]# ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:1Nwxc1WS4aVd9uaIBoT6bBwLx/yd5zvVsbaFw+eoK9k root@fedora
The key's randomart image is:
+---[RSA 4096]-----+
|      .o...ooB|
|      .+  .o.Bo|
|      +o . o + +|
|      o.= . + = |
|      *S+ .oo..=|
|      * ..o =o=|
|      .  o o.*o|
|      o E +..|
|      .oo.o |
+-----[SHA256]-----+
[root@fedora ~]# ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/root/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_ed25519
Your public key has been saved in /root/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:YujgQbrmaVKuPm8z1R2HTaSRhLRefXfaWayR5pAgBaI root@fedora
The key's randomart image is:
+--[ED25519 256]--+
|      ..+o=+o  |
|      o.o+o . . o |
|      . E. o+. .o.+o|
|      o ...o o. .+oo|
|      . o ..= S   .oo |
|      o.+ + o      |
|      .+. o        |
|      +oo+         |
|      ==+.o        |
+-----[SHA256]-----+
[root@fedora ~]#

```

Рис. 3.4: Создание SSH-ключа

Генерирую pgr ключи, используя команду `gpg --full-generate-key` (рис. 3.5).

```
foot
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: SVPavliuchenkov
Адрес электронной почты: thefrankleen@gmail.com
Примечание:
Вы выбрали следующий идентификатор пользователя:
    "SVPavliuchenkov <thefrankleen@gmail.com>"

Сменить (N)Имя, (C)Примечание, (E)Адрес; (O)Принять/(Q)Выход? o
Необходимо получить много случайных чисел. Желательно, чтобы Вы
в процессе генерации выполняли какие-то другие действия (печать
на клавиатуре, движения мыши, обращения к дискам); это даст генератору
случайных чисел больше возможностей получить достаточное количество энтропии.
Необходимо получить много случайных чисел. Желательно, чтобы Вы
в процессе генерации выполняли какие-то другие действия (печать
на клавиатуре, движения мыши, обращения к дискам); это даст генератору
случайных чисел больше возможностей получить достаточное количество энтропии.
gpg: /root/.gnupg/trustdb.gpg: создана таблица доверия
gpg: создан каталог '/root/.gnupg/openpgp-revocs.d'
gpg: сертификат отзыва записан в '/root/.gnupg/openpgp-revocs.d/FA147584C5687AFB1B2E
B9FC9A529D06C7A364C7.rev'.
открытый и секретный ключи созданы и подписаны.

pub   rsa4096 2024-06-21 [SC]
      FA147584C5687AFB1B2EB9FC9A529D06C7A364C7
uid                               SVPavliuchenkov <thefrankleen@gmail.com>
sub   rsa4096 2024-06-21 [E]
```

Рис. 3.5: Создание PGP-ключа

Также выбираю опции, а именно: тип RSA and RSA; размер 4096; срок действия не истекает никогда

Получаю такие ключи.

```
[root@fedora ~]# 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: 1 подписанных: 0 доверие: 0-, 0q, 0n, 0m, 0f, 1u
[keyboard]
-----
sec   rsa4096/9A529D06C7A364C7 2024-06-21 [SC]
      FA147584C5687AFB1B2EB9FC9A529D06C7A364C7
uid                               [ абсолютно ] SVPavliuchenkov <thefrankleen@gmail.com>
ssb   rsa4096/6B79B908D394981E 2024-06-21 [E]

[root@fedora ~]#
```

Рис. 3.6: Полученные PGP-ключи


Копирую отпечаток ключа, и захожу на github, чтобы добавить новый ключ.

SSH keys

[New SSH key](#)

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

Authentication keys

**Laptop**
SHA256:9HIEG0q50qHItt9KkG2+JS8W2gcX9L9AIHTtYtViHjI
Added on Dec 22, 2023
Last used within the last 6 months — Read/write
[Delete](#)

Check out our guide to [connecting to GitHub using SSH keys](#) or troubleshoot [common SSH problems](#).

GPG keys

[New GPG key](#)

There are no GPG keys associated with your account.

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

Рис. 3.7: Настройки github

Вношу отпечаток своего ключа на сайт.

Add new GPG key

Title

Fedora

Key

```
-----BEGIN PGP PUBLIC KEY BLOCK-----  
iQI2BCABCAAgFiEE+hr1hMVoevsbLrn8mlKdBsejZMcFamZ1wWoCH  
QAACgkQmlKd  
BsejZMfR2hAAwsJS7vpLeapLc4qqMdsT0s8+4O1mCiW7BGYNmuP0a  
x3lEjjBrWJZ  
59TyI7kkpYFQ6FGW/6fz0RKi4xK+QAAoNhk48ixqvRWwwwxjlzYR+Vtt8  
HVwtNrG  
GCLAIHH62hoBj5/pzb36y/  
MtG2fFiRnt+4usfCcrTIdc7o58qyzJ7CZ1YvNxDwyn
```

Add GPG key


Рис. 3.8: Заполнение ключа

У меня получилось добавить новый ключ.

GPG keys

[New GPG key](#)

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


GPG

fedora
Email address: thefrankleen@gmail.com
Key ID: 9A529D06C7A364C7
Subkeys: 6B79B908D394981E
Added on Jun 21, 2024
[Delete](#)

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

Vigilant mode

Рис. 3.9: Новый ключ

Указываю Git применять введенную почту при подписи коммитов.

```
[root@fedora ~]# git config --global user.signingkey 9A529D06C7A364C7
[root@fedora ~]# git config --global commit.gpgsign true
[root@fedora ~]# git config --global gpg.program $(which gpg2)
[root@fedora ~]# gh
```

Рис. 3.10: Настраиваю подписи коммитов git.

Авторизуюсь используя gh auth login

```

[root@fedora ~]# 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? /root/.ssh/id_rsa.pub
? Title for your SSH key: fedora2
? How would you like to authenticate GitHub CLI? Paste an authentication token
Tip: You can generate a Personal Access Token here https://github.com/settings
The minimum required scopes are 'repo', 'read:org', 'admin:public_key'.
? Paste your authentication token: *****
- gh config set -h github.com git_protocol ssh
/ Configured git protocol
! Authentication credentials saved in plain text
/ Uploaded the SSH key to your GitHub account: /root/.ssh/id_rsa.pub
/ Logged in as serapshi

```

Рис. 3.11: Авторизация gh

Создаю репозиторий курса на основе шаблона

```
[root@fedora ~]# mkdir -p ~/work/study2023-2024/OS
[root@fedora ~]# cd ~/work/study2023-2024/OS/
[root@fedora OS]# gh repo create study2023-2024_os-intro --template=yamadharma/course-directory-student-template --public
Created repository serapshi/study2023-2024_os-intro on GitHub
https://github.com/serapshi/study2023-2024_os-intro
[root@fedora OS]# git clone --recursive git@github.com:serapshi/study2023-2024_os-intro.git os-intro
Клонирование в «os-intro»...
The authenticity of host 'github.com (140.82.121.3)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvV6TuJJhpZisF/zLDA0zPMSvHdK4UvCOqU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
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), 18.61 КиБ | 18.61 МБ/с, готово.
Определение изменений: 100% (1/1), 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 «/root/work/study2023-2024/OS/os-intro/template/presentation»...
remote: Enumerating objects: 95, done.
remote: Counting objects: 100% (95/95), done.
remote: Compressing objects: 100% (67/67), done.
remote: Total 95 (delta 34), reused 87 (delta 26), pack-reused 0
Получение объектов: 100% (95/95), 96.99 КиБ | 1003.00 КиБ/с, готово.
Определение изменений: 100% (34/34), готово.
Клонирование в «/root/work/study2023-2024/OS/os-intro/template/report»...
remote: Enumerating objects: 126, done.
remote: Counting objects: 100% (126/126), done.
remote: Compressing objects: 100% (87/87), done.
remote: Total 126 (delta 52), reused 108 (delta 34), pack-reused 0
Получение объектов: 100% (126/126), 335.80 КиБ | 1.78 МБ/с, готово.
Определение изменений: 100% (52/52), готово.
Submodule path 'template/presentation': checked out '40a1761813e197d00e8443ff1ca72c60a304f24c'
Submodule path 'template/report': checked out '7c31ab8e5dfa8cdb2d67caeb8a19ef8028ced88e'
[root@fedora OS]#
```

Рис. 3.12: Создание репозитория курса на основе шаблона

Удаляю лишние файлы(рис. 3.13).

```
[root@fedora os-intro]# rm package.json
rm: удалить обычный файл 'package.json'? yes
[root@fedora os-intro]# ls
CHANGELOG.md  COURSE  Makefile  README.git-flow.md  template
config        LICENSE  README.en.md  README.md
[root@fedora os-intro]#
```

Рис. 3.13: Настройка каталога курса

Создаю необходимые каталоги.

```
[root@fedora os-intro]# echo os-intro > COURSE
[root@fedora os-intro]# make
Usage:
  make <target>

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

Рис. 3.14: Создание необходимых каталогов

Отправляю файлы на сервер.

```
[root@fedora os-intro]# git add .
[root@fedora os-intro]# git commit -am 'feat(main): make course structure'
[master af19dda] feat(main): make course structure
2 files changed, 1 insertion(+), 14 deletions(-)
delete mode 100644 package.json
[root@fedora os-intro]# git push
Перечисление объектов: 5, готово.
Подсчет объектов: 100% (5/5), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (2/2), готово.
Запись объектов: 100% (3/3), 955 байтов | 191.00 КиБ/с, готово.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:serapshi/study2023-2024_os-intro.git
   29bef75..af19dda  master -> master
[root@fedora os-intro]#
```

Рис. 3.15: Использование

4 Выводы

Я подготовил git репозиторий для последующей работы по этому курсу и освежил навыки работы по работе с git.

5 Контрольные вопросы

Что такое системы контроля версий (VCS) и для решения каких задач они предназначены?

Это ПО для отслеживания изменений в файлах. Оно может быть использовано, как в разработке больших систем, так и небольших программ. Объясните следующие понятия VCS и их отношения: хранилище, commit, история, рабочая копия.

Что представляют собой и чем отличаются централизованные и децентрализованные VCS? Приведите примеры VCS каждого вида.

Централизованные хранят все данные на одном сервере, как Яндекс. А децентрализованные такие, как git хранят часть данных на сервере, а другие у самих разработчиков.

Опишите действия с VCS при единоличной работе с хранилищем.

После внесения изменений, пользователь размещает новую версию в хранилище.

Опишите порядок работы с общим хранилищем VCS.

Системы контроля версий также могут обеспечивать дополнительные, более гибкие функциональные возможности. Например, они могут поддерживать работу с несколькими версиями одного файла, сохраняя общую историю изменений до точки ветвления версий и собственные истории изменений каждой ветви. Кроме того, обычно доступна информация о том, кто из участников, когда и какие изменения вносил.

Каковы основные задачи, решаемые инструментальным средством git?

Ветвление, слияние и управление репозиторием.

Назовите и дайте краткую характеристику командам git.

git add - направляет элемент к загрузке на сервер, git push - загружает элементы на сервер

Приведите примеры использования при работе с локальным и удалённым репозиториями.

При работе с локальным: git add .; git commit 'feat(main): commit'; git push. При работе с удаленным: git clone --recursive (URL).

Что такое и зачем могут быть нужны ветви (branches)?

Branch(ветка) - это ответвление, от целого репозитория, где можно заниматься разработкой определенных функций. Для удобства разработки и/или исправления ошибок.

Как и зачем можно игнорировать некоторые файлы при commit?

Во время работы над проектом так или иначе могут создаваться файлы, которые не требуется добавлять в последствии в репозиторий. Например, временные файлы, создаваемые редакторами, или объектные файлы, создаваемые компиляторами. Можно прописать шаблоны игнорируемых при добавлении в репозиторий типов файлов в файл .gitignore с помощью сервисов