

Лабораторная работа №1

Научное программирование

Леонтьева К. А., НПМмд-02-23

21 сентября 2023

Российский университет дружбы народов

Москва, Россия

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

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.

- Вручную установили программное обеспечение git-flow

```
[kaleontjeva@kaleontjeva raw.github.com]$ cd /tmp
[kaleontjeva@kaleontjeva tmp]$ wget -no-check-certificate -q https://raw.githubusercontent.com/petervanderdoes/gitflow/develop/contrib/gitflow-installer.sh
[kaleontjeva@kaleontjeva tmp]$ chmod +x gitflow-installer.sh
[kaleontjeva@kaleontjeva tmp]$ sudo ./gitflow-installer.sh install stable
### git-flow no-make installer ###
Installing git-flow to /usr/local/bin
Cloning repo from GitHub to gitflow
Клонирование в «gitflow»...
remote: Enumerating objects: 4270, done.
remote: Total 4270 (delta 0), reused 0 (delta 0), pack-reused 4270
Получение объектов: 100% (4270/4270), 1.74 МБ | 675.00 КБ/с, готово.
Определение изменений: 100% (2533/2533), готово.
Уже обновлено.
Ветка «master» отслеживает внешнюю ветку «master» из «origin».
Переключено на новую ветку «master»
install: создание каталога '/usr/local/share/doc'
install: создание каталога '/usr/local/share/doc/gitflow'
install: создание каталога '/usr/local/share/doc/gitflow/hooks'
'gitflow/git-flow' -> '/usr/local/bin/git-flow'
'gitflow/git-flow-init' -> '/usr/local/bin/git-flow-init'
'gitflow/git-flow-feature' -> '/usr/local/bin/git-flow-feature'
'gitflow/git-flow-bugfix' -> '/usr/local/bin/git-flow-bugfix'
'gitflow/git-flow-hotfix' -> '/usr/local/bin/git-flow-hotfix'
'gitflow/git-flow-release' -> '/usr/local/bin/git-flow-release'
'gitflow/git-flow-support' -> '/usr/local/bin/git-flow-support'
'gitflow/git-flow-version' -> '/usr/local/bin/git-flow-version'
'gitflow/gitflow-common' -> '/usr/local/bin/gitflow-common'
'gitflow/gitflow-shFlags' -> '/usr/local/bin/gitflow-shFlags'
'gitflow/git-flow-config' -> '/usr/local/bin/git-flow-config'
'gitflow/hooks/filter-flow-hotfix-finish-tag-message' -> '/usr/local/share/doc/gitflow/hooks/filter-flow-hotfix-finish-tag-message'
'gitflow/hooks/filter-flow-hotfix-start-version' -> '/usr/local/share/doc/gitflow/hooks/filter-flow-hotfix-start-version'
'gitflow/hooks/filter-flow-release-branch-tag-message' -> '/usr/local/share/doc/gitflow/hooks/filter-flow-release-branch-tag-message'
```

Figure 1: Рис.1: Установка git-flow

Ход выполнения лабораторной работы

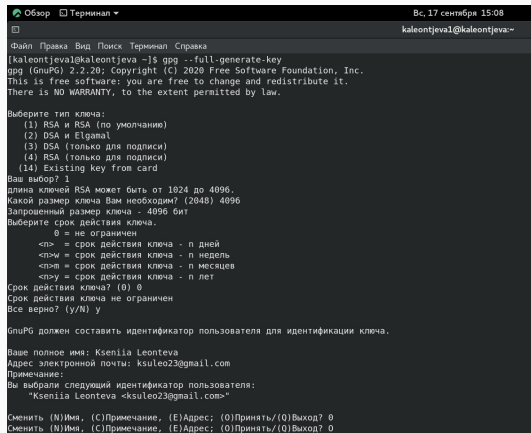
- Осуществили базовую настройку git

```
[kaleontjeval@kaleontjeva ~]$ git config --global user.name "Kseniia Leonteva"
[kaleontjeval@kaleontjeva ~]$ git config --global user.email "ksuleo23@gmail.com"
[kaleontjeval@kaleontjeva ~]$ git config --global core.quotepath false
[kaleontjeval@kaleontjeva ~]$ git config --global init.defaultBranch master
[kaleontjeval@kaleontjeva ~]$ git config --global core.autocrlf input
[kaleontjeval@kaleontjeva ~]$ git config --global core.safecrlf warn
[kaleontjeval@kaleontjeva ~]$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/kaleontjeval/.ssh/id_rsa): /home/kaleontjeval/.ssh/id_rsa
/home/kaleontjeval/.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 /home/kaleontjeval/.ssh/id_rsa.
Your public key has been saved in /home/kaleontjeval/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:YiGPwd7T7zdgbRl7LpYh3Bnvt0XceggjWs9Jyc6JQ kaleontjeval@kaleontjeva
The key's random image is:
+---[RSA 4096]-----+
|
|  . . . . .
|  =o. . . o
| o +.+oS = . o
| o o++ % * .
|  ..oE.X .
| oo+Oo+ . .
| . o*O.. .
+----[SHA256]-----+
```

Figure 2: Рис.2: Базовая настройка git

Ход выполнения лабораторной работы

- Сгенерировали ключ PGP и выбрали необходимые опции. Затем скопировали ключ и добавили его на GitHub



```
Обзор Терминал
kaleontjeval@kaleontjeva:~$ gpg --full-generate-key
gpg (GnuPG) 2.2.20; Copyright (C) 2020 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.

Выберите тип ключа:
(1) RSA и RSA (по умолчанию)
(2) DSA и ElGamal
(3) DSA (только для подписи)
(4) RSA (только для подписи)
(14) Existing key from card
Ваш выбор? 1
длина ключей RSA может быть от 1024 до 4096.
Какой размер ключа Вам необходим? (2048) 4096
Запрошенный размер ключа - 4096 бит
Выберите срок действия ключа.
0 = не ограничен
<p> = срок действия ключа - п дней
<p>w = срок действия ключа - п недель
<p>m = срок действия ключа - п месяцев
<p>y = срок действия ключа - п лет
Срок действия ключа? (0) 0
Срок действия ключа не ограничен
Все верно? (y/N) y

GnuPG должен составить идентификатор пользователя для идентификации ключа.

Ваше полное имя: Ksenia Leonteva
Адрес электронной почты: ksuleo23@gmail.com
Примечание:
Вы выбрали следующий идентификатор пользователя:
"Ksenia Leonteva <ksuleo23@gmail.com>"

Сменить (N)Имя, (C)Примечание, (E)Адрес; (O)Принять/(Q)Выход? 0
Сменить (N)Имя, (C)Примечание, (E)Адрес; (O)Принять/(Q)Выход? 0
```

Figure 3: Рис.3: Генерация ключа PGP

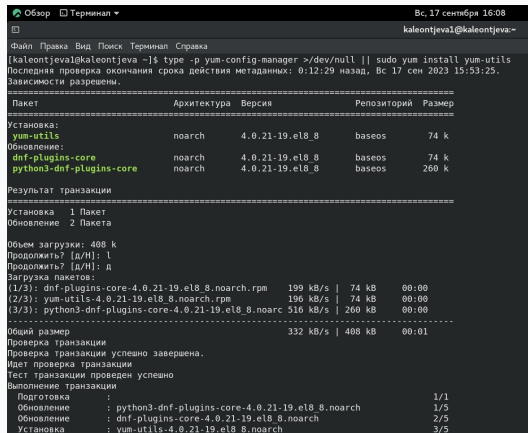
- Настроили автоматические подписи коммитов git

```
[kaleontjeval@kaleontjeva ~]$ git config --global user.signingkey ksuleo23@gmail.com
[kaleontjeval@kaleontjeva ~]$ git config --global commit.gpgsign true
[kaleontjeval@kaleontjeva ~]$ git config --global gpg.program $(which gpg2)
[kaleontjeval@kaleontjeva ~]$
```

Figure 4: Рис.4: Настройка автоматических подписей коммитов git

Ход выполнения лабораторной работы

- Установили и настроили gh



```

Обзор Терминал
kaleontjeval@kaleontjeva:~
[кaleontjeval@kaleontjeva ~]$ type -p yum-config-manager >/dev/null || sudo yum install yum-utils
Последняя проверка окончания срока действия метаданных: 0:12:29 назад, Вс 17 сен 2023 15:53:25.
Зависимости разрешены.
=====
Пакет                Архитектура  Версия          Репозиторий  Размер
=====
Установка:
yum-utils             noarch       4.0.21-19.el8_8 baseos       74 k
Обновление:
dnf-plugins-core      noarch       4.0.21-19.el8_8 baseos       74 k
python3-dnf-plugins-core noarch       4.0.21-19.el8_8 baseos      260 k
=====
Результат транзакции
=====
Установка 1 Пакет
Обновление 2 Пакета

Объем загрузки: 408 k
Продолжить? [д/Н]: л
Продолжить? [д/Н]: д
Загрузка пакетов:
(1/3): dnf-plugins-core-4.0.21-19.el8_8.noarch.rpm 199 kB/s | 74 kB 00:00
(2/3): yum-utils-4.0.21-19.el8_8.noarch.rpm      196 kB/s | 74 kB 00:00
(3/3): python3-dnf-plugins-core-4.0.21-19.el8_8.noarch 516 kB/s | 260 kB 00:00
-----
Общий размер                332 kB/s | 408 kB 00:01
Проверка транзакции
Проверка транзакции успешно завершена.
Идет проверка транзакции
Тест транзакции проведен успешно
Выполнение транзакции
Подготовка :                               1/1
Обновление : python3-dnf-plugins-core-4.0.21-19.el8_8.noarch 1/5
Обновление : dnf-plugins-core-4.0.21-19.el8_8.noarch      2/5
Установка  : yum-utils-4.0.21-19.el8_8.noarch             3/5

```

Figure 5: Рис.5: Установка gh

- Создали репозиторий курса на основе шаблона и настроили каталог курса

```
[kaleontjeval@kaleontjeva ~]$ cd ~/work/study
[kaleontjeval@kaleontjeva study]$ mkdir -p 2023-2024/"Научное программирование"
[kaleontjeval@kaleontjeva study]$ cd 2023-2024/"Научное программирование"
[kaleontjeval@kaleontjeva Научное программирование]$ gh repo create study_2023-2024_sc
iprog-intro--template=yamadharm/course-directory-student-template --public
HTTP 404: Not Found (https://api.github.com/users/study_2023-2024_sciprog-intro--templ
ate=yamadharm)
[kaleontjeval@kaleontjeva Научное программирование]$ gh repo create study_2023-2024_sc
iprog--template=yamadharm/course-directory-student-template --public
HTTP 404: Not Found (https://api.github.com/users/study_2023-2024_sciprog--template=ya
madharma)
[kaleontjeval@kaleontjeva Научное программирование]$ gh repo create study_2023-2024_sc
iprog --template=yamadharm/course-directory-student-template --public
✓ Created repository Ksenia-Leonteva/study_2023-2024_sciprog on GitHub
```

Figure 6: Рис.6: Создание репозитория курса

- В ходе выполнения данной лабораторной работы я изучила идеологию и применение средств контроля версий, а также освоила умения по работе с git.