

Отчет по лабораторной работе №2.

Работу выполнил Шеожев Аслан Аскерович.

Цель работы:

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

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.

Ход работы:

1. Ознакомился с основными командами git и установил все необходимое ПО (get-flow, gh). Настроил github. Создал учетную запись и заполнил основные данные.
2. Задал имя и email владельца репозитория, настроил utf-8 в выводе сообщений git, задал имя начальной ветки (будем называть её master). Создал ключи ssh по алгоритму rsa с ключём размера 4096 бит и по алгоритму ed25519.

```
[aasheozhev@fedora tmp]$ git config --global user.name "Aslan Sheozhev"
[aasheozhev@fedora tmp]$ git config --global user.email "1032216531@rudn.ru"
[aasheozhev@fedora tmp]$ git config --global core.quotepath false
[aasheozhev@fedora tmp]$ git config --global init.defaultBranch master
[aasheozhev@fedora tmp]$ git config --global core.autocrlf input
[aasheozhev@fedora tmp]$ git config --global core.safecrlf warn
[aasheozhev@fedora tmp]$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/aasheozhev/.ssh/id_rsa): 123
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
```

width=100% } Рис. 1 (ввод основных сведений о программисте, а также настройка –utf8)

3. Создал ключи pgp. Тип – RSA and RSA. Размер – 4096. Срок действия не истекает никогда

```

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) 0
Срок действия ключа не ограничен
Все верно? (y/N) y

GnuPG должен составить идентификатор пользователя для идентификации ключа.

Ваше полное имя: Aslan
Адрес электронной почты: 1032216531@rudn.ru
Примечание: -
Вы выбрали следующий идентификатор пользователя:
  "Aslan (-) <1032216531@rudn.ru>"

Сменить (N)Имя, (C)Примечание, (E)Адрес; (O)Принять/(Q)Выход? E
Адрес электронной почты: aslan.sheudzhen@mail.ru
Вы выбрали следующий идентификатор пользователя:
  "Aslan (-) <aslan.sheudzhen@mail.ru>"

Сменить (N)Имя, (C)Примечание, (E)Адрес; (O)Принять/(Q)Выход? C
Примечание:
Вы выбрали следующий идентификатор пользователя:
  "Aslan <aslan.sheudzhen@mail.ru>"

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: /home/aasheozhev/.gnupg/trustdb.gpg: создана таблица доверия
gpg: ключ C060C48E60F719A2 помечен как абсолютно доверенный
gpg: создан каталог '/home/aasheozhev/.gnupg/openpgp-revocs.d'
gpg: сертификат отзыва записан в '/home/aasheozhev/.gnupg/openpgp-revocs.d/F7BCD898D921242E3EDB87F2C060C48E60F719A2.rev'.
открытый и секретный ключи созданы и подписаны.

pub  rsa4096 2022-04-23 [SC]
    F7BCD898D921242E3EDB87F2C060C48E60F719A2
uid                               Aslan <aslan.sheudzhen@mail.ru>
sub  rsa4096 2022-04-23 [E]

```



{ width=100% } Рис. 2 (создал ключи pgp)

- Добавил pgr и ssh ключи в github.

SSH keys

[New SSH key](#)

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


 SSH	ssh-rsa AAAAB3NzaC1yc2EAAA SHA256: EFT19NulwTc0x0Na0rtSXb458Cr0zohmTQGf/p3A0fo0 Added on 23 Apr 2022 Never used — Read/write	Delete
 SSH	GitHub CLI SHA256: E59yes3mR73KQm5INLaguPKfLnX1cxrhtqwyDfnH3/w Added on 23 Apr 2022 by GitHub CLI Last used within the last week — Read/write	Delete

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

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	Email address: <code>aslan.sheudzhen@mail.ru</code> Key ID: C060C48E60F719A2 Subkeys: 7E30A52C66F077E5 Added on 23 Apr 2022	Delete
--	---	------------------------

{ width=100% } Рис. 3 (введенные ключи на github)

- Используя введенный email, указал Git применять его при подписи коммитов.

```
[aasheozhev@fedora tmp]$ gpg --armor --export C060C48E60F719A2 | xclip -sel clip
[aasheozhev@fedora tmp]$ git config --global user.signingkey C060C48E60F719A2
[aasheozhev@fedora tmp]$ git config --global commit.gpgsign true
[aasheozhev@fedora tmp]$ git config --global gpg.program $(which gpg2)
```

{ width=100% }

Рис. 4 (для подписи коммитов)

- Приступил к настройке gh.
- Создал шаблон рабочего пространства, по примеру из условия лабораторной работы. А после настроил каталог курса.

Aslan Sheozhev feat(main): make course structure			af733e3 36 minutes ago	🕒 2 commits
📁 config	Initial commit			3 hours ago
📁 labs	feat(main): make course structure			36 minutes ago
📁 project-personal	feat(main): make course structure			36 minutes ago
📁 template	Initial commit			3 hours ago
📄 .gitattributes	Initial commit			3 hours ago
📄 .gitignore	Initial commit			3 hours ago
📄 .gitmodules	Initial commit			3 hours ago
📄 LICENSE	Initial commit			3 hours ago
📄 Makefile	Initial commit			3 hours ago
📄 README.en.md	Initial commit			3 hours ago
📄 README.git-flow.md	Initial commit			3 hours ago
📄 README.md	Initial commit			3 hours ago
📄 structure	feat(main): make course structure			36 minutes ago

{ width=100% } Рис. 5 (результат на странице репозитория в github)

Вывод:

Во второй лабораторной работе я изучил идеологию и применение средств контроля версий, освоил умения по работе с git.

Контрольные вопросы:

1. Что такое системы контроля версий (VCS) и для решения каких задач они предназначены?
 - Система контроля версий (VCS) нужны для работы нескольких людей над одним проектом, совместная работа путем изменения файлов в репозитории.
2. Объясните следующие понятия VCS и их отношения: хранилище, commit, история, рабочая копия.
 - хранилище – это пространство для хранения файлов;
 - commit – команда для записи индексированных изменений в репозитория;
 - история – в ней расположены все коммиты, по которым можно отследить автора, дату и сообщение;
 - рабочая копия – все файлы, кроме .git/ это рабочая копия.
3. Что представляют собой и чем отличаются централизованные и децентрализованные VCS? Приведите примеры VCS каждого вида.
 - Централизованные системы контроля версий – проект и его файлы на 1 сервер, а децентрализованные системы контроля версия – при копировании данных удаленного репозитория, происходит полное копирование данных в локальный репозиторий.

4. Опишите действия с VCS при единоличной работе с хранилищем.
 - Создание репозиторий, добавление в них файлов, фиксация с помощью коммитов, изменение файлов репозитория.
5. Опишите порядок работы с общим хранилищем VCS.
 - Создание репозитория, добавление файлов в него, фиксация коммитами, изменение файлов репозитория, ожидание проверки коммитов.
6. Каковы основные задачи, решаемые инструментальным средством git?
 - Систематизация и возможность разработки параллельно.
7. Назовите и дайте краткую характеристику командам git.
 - Создание репозитория (git init), клонирование (git clone), добавление изменений в индекс (git add), коммиты (git commit), удаление файла (git rm), удаление изменений из индекса (git reset).
8. Приведите примеры использования при работе с локальным и удалённым репозиториями.
 - Для написания черновика по лабораторным работам я могу использовать репозитории.
9. Что такое и зачем могут быть нужны ветви (branches)?
 - Ветки служат для реализации параллельного программирования.
10. Как и зачем можно игнорировать некоторые файлы при commit?
 - Игнорирование нужно для неотслеживания файлов служебного типа.