

Лабораторная работа №2. Управление версиями.

Подготовил: Королев Адам Маратович

Группа: НПИбд-02-21

Номер студенческого билета: 1032217060

Цель работы:

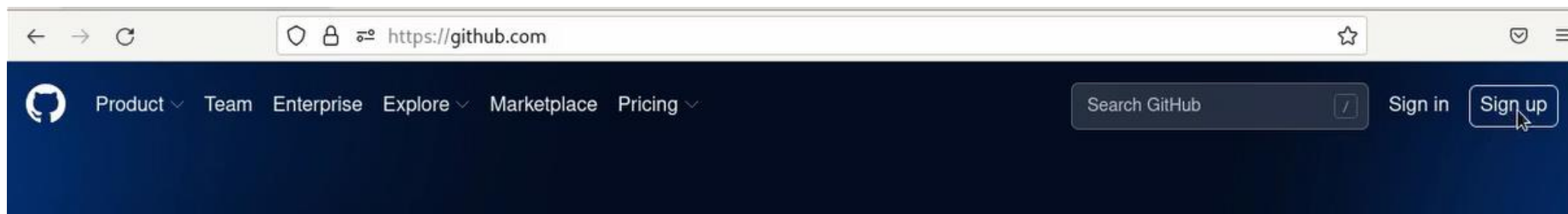
- Изучить идеологию и применение средств контроля версий.
- Освоить умения по работе с git.

Задание:

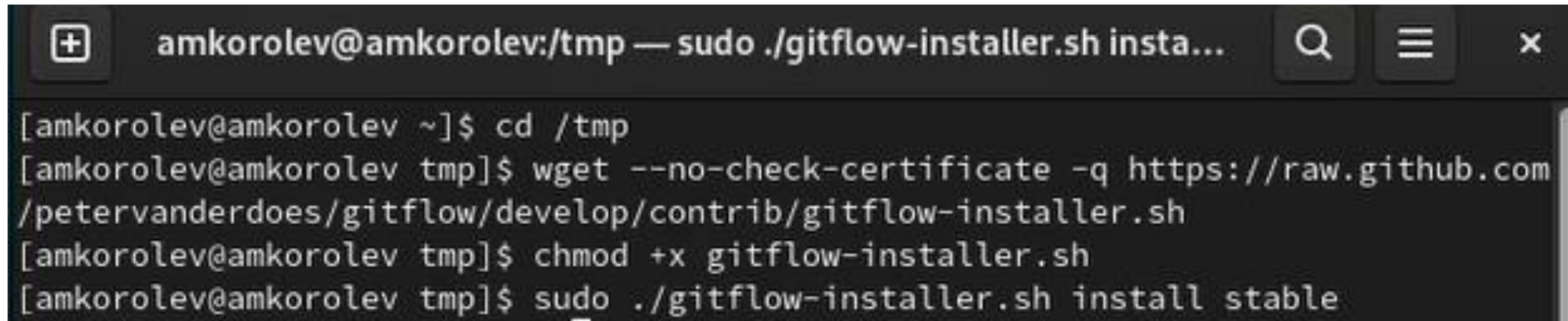
- Создать базовую конфигурацию для работы с git.
- Создать ключ SSH.
- Создать ключ PGP.
- Настроить подписи git.
- Создать локальный каталог для выполнения заданий по предмету.

Выполнение лабораторной работы

- Необходимо создать учетную запись на github.com



- Установить git-flow в Fedora Linux. Устанавливается оно вручную, так как оно удалено из репозитория.



```
amkorolev@amkorolev:/tmp — sudo ./gitflow-installer.sh insta...
[amkorolev@amkorolev ~]$ cd /tmp
[amkorolev@amkorolev tmp]$ wget --no-check-certificate -q https://raw.githubusercontent.com/petervanderdoes/gitflow/develop/contrib/gitflow-installer.sh
[amkorolev@amkorolev tmp]$ chmod +x gitflow-installer.sh
[amkorolev@amkorolev tmp]$ sudo ./gitflow-installer.sh install stable
```

- Необходимо установить gh в Fedora Linux

```
[amkorolev@amkorolev tmp]$ sudo dnf install gh
```

- Необходимо произвести базовую настройку git. То есть, задать имя и email владельца репозитория.

```
[amkorolev@amkorolev tmp]$ git config --global user.name "Adam Korolev"  
[amkorolev@amkorolev tmp]$ git config --global user.email "amkorolev@amkorolev.com"
```

- Настроим utf-8 в выводе сообщений git.

```
[amkorolev@amkorolev tmp]$ git config --global core.quotePath false
```


Настроим верификацию и подписание КОММИТОВ git:

Зададим имя начальной ветки (будем называть ее master).

```
[amkorolev@amkorolev tmp]$ git config --global init.defaultBranch master
```

- Установим параметр autocrlf.

```
[amkorolev@amkorolev tmp]$ git config --global core.autocrlf input
```

Установим параметр safecrlf.

```
[amkorolev@amkorolev tmp]$ git config --global core.safecrlf warn
```

- По алгоритму rsa создадим ключ ssh с размером 4096 бит.

```
[amkorolev@amkorolev ~]$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/amkorolev/.ssh/id_rsa): /home/amkorolev/.ssh/id_rsa
Created directory '/home/amkorolev/.ssh'.
```

- По алгоритму ed25519 создадим ключ ssh.

```
[amkorolev@amkorolev ~]$ ssh-keygen -t ed25519
```

Создадим ключ ргр путем генерации.

- Из предложенных опций выбираем:
- Тип RSA and RSA;
- Размер 4096;
- Устанавливаем срок действия, параметр 0 (не истекает никогда);
- Далее GPG запросит личную информацию, которая сохранится в ключе:
- Имя;
- Адрес электронной почты;
- Email (необходимо ввести email, используемый на GitHub)
- Комментарий. Можно оставить пустым.

```
[amkorolev@amkorolev ~]$ gpg --full-generate-key
```

```
gpg: создан каталог '/home/amkorolev/.gnupg'
gpg: создан щит с ключами '/home/amkorolev/.gnupg/pubring.kbx'
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
Ваш выбор? 1
длина ключей RSA может быть от 1024 до 4096.
Какой размер ключа Вам необходим? (3072) 4096
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
```

GnuPG должен составить идентификатор пользователя для идентификации ключа.

Ваше полное имя: Adam Korolev

Адрес электронной почты:

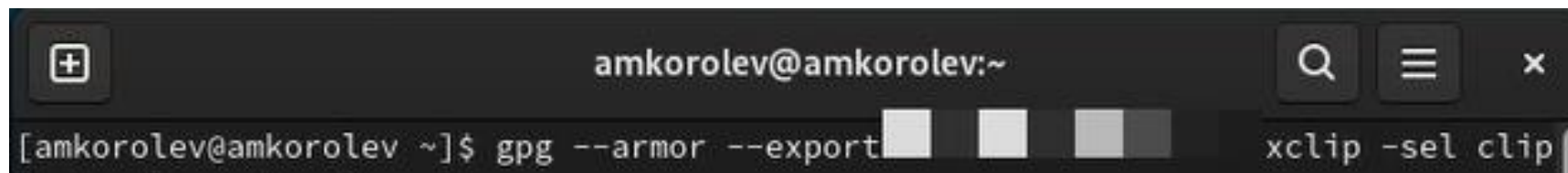
Сменить (N)Имя, (C)Примечание, (E)Адрес; (O)Принять/(Q)Выход? O

- Выводим список ключей и копируем отпечаток приватного ключа.

```
[amkorolev@amkorolev ~]$ gpg --list-secret-keys --keyid-format LONG
```

```
-----  
sec          [redacted]  
uid          [redacted]  
ssb          [redacted]
```


- Скопируем наш сгенерированный PGP ключ в буфер обмена.



```
amkorolev@amkorolev:~  
[amkorolev@amkorolev ~]$ gpg --armor --export xclip -sel clip
```

The image shows a terminal window with a dark background. The title bar at the top reads 'amkorolev@amkorolev:~'. On the left of the title bar is a square button with a white plus sign. On the right are three buttons: a magnifying glass (search), a hamburger menu (three horizontal lines), and a close button (an 'x'). The terminal prompt is '[amkorolev@amkorolev ~]\$'. The command 'gpg --armor --export' is entered, followed by a space and the start of another command 'xclip -sel clip'. The cursor is at the end of the second command.

- Перейдем в настройки GitHub, нажмем на кнопку New GPG key и вставим полученный ключ в поле ввода.

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](#) .

GPG keys / Add new

Key

Add GPG key

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

```
[amkorolev@amkorolev ~]$ git config --global user.signingkey  
[amkorolev@amkorolev ~]$ git config --global commit.gpgsign true  
[amkorolev@amkorolev ~]$ git config --global gpg.program $(which gpg2)
```

- Произведем авторизацию в GitHub.

```
[amkorolev@amkorolev ~]$ gh auth login
```

- Создадим шаблон рабочего пространства.

```
[amkorolev@amkorolev ~]$ mkdir -p ~/work/study/2021-2022/"Операционные системы"
[amkorolev@amkorolev ~]$ cd ~/work/study/2021-2022/Операционные\ системы/
[amkorolev@amkorolev Операционные системы]$ gh repo create study_2021-2022_os-intro --template=yamadharma/course-directory-student-template --public
✓ Created repository ALibA0/study_2021-2022_os-intro on GitHub
[amkorolev@amkorolev Операционные системы]$ git clone --recursive git@github.com:ALibA0/study_2021-2022_os-intro.git os-intro
```

- Перейдем в каталог курса.

```
[amkorolev@amkorolev Операционные системы]$ cd ~/work/study/2021-2022/Операционн  
ые\ системы/os-intro/
```

- Удалим лишние файлы.

```
[amkorolev@amkorolev os-intro]$ rm package.json
```

- Создадим необходимые каталоги.

```
[amkorolev@amkorolev os-intro]$ make COURSE=os-intro
```


- Отправим файлы на сервер.

```
[amkorolev@amkorolev os-intro]$ git add .  
[amkorolev@amkorolev os-intro]$ git commit -am 'feat(main): make course structur  
e'
```

```
[amkorolev@amkorolev os-intro]$ git push
```

Выводы:

- В процессе выполнения задания были приобретены умения по работе с git. Была изучена идеология и применения средств контроля версий.