

# Лабораторная работа №1

---

Подмогильный Иван Александрович - студент группы НПМмд-02-22

17.09.2022

## Работа в Git

---

Умение пользоваться Git

## Цель выполнения лабораторной работы

Освоить на практике использование Git, Github, git flow

- 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.

## Результаты выполнения лабораторной работы. Установил git-flow в Ubuntu

```
Activities Terminal - C:\Users\rajeev...
git@rajeevpc:~$ sudo apt install git-flow
[sudo] password for rajeev:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libglib2.0-0
Use 'dpkg --get-selections' to remove them.
The following new packages will be installed:
  git-flow
0 upgraded, 1 newly installed, 0 to remove and 168 not upgraded.
Need to get 39.8 kB of archives.
After this operation, 104.8 kB of additional disk space will be used.
Get:1 /tmp/apt-mirror.xubuntu.com/ubuntu focal/universe amd64 git-flow all 1.12.3-1 [39.8 kB]
debconf: warning: unable to write user configuration to /etc/gitflow.conf
Fetched 39.8 kB in 1s (27.7 kB/s)
Selecting previously unselected package git-flow.
(Reading database ... 86895 files and directories currently installed.)
Preparing to unpack .../git-flow_1.12.3-1_all.deb ...
Unpacking git-flow (1.12.3-1) ...
Setting up git-flow (1.12.3-1) ...
Usage: git flow <command>
Available subcommands are:
init      Initialize a new git repo with support for the branching model.
feature   Manage your feature branches.
hotfix    Manage your hotfix branches.
release   Manage your release branches.
branch    Manage your branch branches.
support   Manage your support branches.
version   Show version information.
config    Manage your git-flow configurations.
log       Show log detailing from base branch.
```

Figure 1: 1



Задал базовую настройку git. Настроил utf-8 в выводе сообщений гит.

Зададим имя начальной ветки (будем называть её `master`). Настроим верификацию и подписание коммитов `git`.

```

root@ubuntu:~# git config --global user.name "John Doe"
root@ubuntu:~# git config --global user.email "johndoe@gmail.com"
root@ubuntu:~# git commit -m "Initial commit"
[master (root-commit) 0d7c8e9] Initial commit
1 file changed, 1 insertion(+), 0 deletions(-)
create mode 160000 README.md
root@ubuntu:~# git push origin master
Counting objects: 1, done.
Writing objects: 1 (160000 bytes) | 160000 bytes/sec, 1.00 MB/s
Compressing objects: 0, done.
Verifying commit: OK
remote: Compressing...
To github.com:john-doe/repo.git
 * [new branch]      master -> master
root@ubuntu:~#
```

---

**ACTIONS**

- `auth:` Authenticate gh and git with Github
- `drawn:` Draw the Repository as the browser
- `consequence:` Commit to and manage your consequences
- `git:` Manage git
- `issue:` Manage Issues
- `pull-request:` Manage pull requests
- `release:` Manage Releases
- `repo:` Manage repositories

**ACTIONS COMMANDS**

- `run:` View details about workflow runs
- `workflow:` View details about Github Actions workflows

---

**ADDITIONAL COMMANDS**

- `alias:` Create command shortcuts
- `auth:` Authenticate Github API request
- `completion:` Generate shell completion scripts
- `config:` Manage configuration for gh
- `extension:` Manage GH extensions
- `gh-key:` Manage GH keys
- `help-gh:` Help about our command
- `search:` Search for repositories, issues, pull requests and users
- `secret:` Manage Github secrets
- `ssh-key:` Manage SSH keys

---

**HELP TOPICS**

- `actions:` Learn about working with Github Actions
- `autocomplete:` Environment variables that can be used with gh
- `installing:` Installing options for 32bit arches and from gh
- `intro:` Introduction about using gh with HTTP
- `reference:` A comprehensive reference of all gh commands

---

**FLAGS**

- `-help` Show help for command
- `-version` Show gh version

---

**EXAMPLES**

- `gh issue create`
- `gh repo clone <owner>`
- `gh pr checkout <id>`

---

**LEARN MORE**

- Use "`gh -command --help`" for more information about a command.
- Read the manual at <https://cli.github.com/manual>.

---

**PERFORMA**

- Show an issue using 'gh issue create' # github.com/CUAACT'

```

github-mali102 ~$ gh config -g github user name "Dore Pedreglay"
github-mali102 ~$ gh config -g github user email "pedreglay@gmail.com"
github-mali102 ~$ gh config -g github core sshpath /etc
github-mali102 ~$ gh config -g github url https://github.com
github-mali102 ~$ gh config -g github core.sshkey id_rsa
github-mali102 ~$ gh config -g github core.sshkey id_rsa
github-mali102 ~$
```

Figure 3: 3



Создайте ключи ssh по алгоритму rsa с ключём размером 4096 бит по алгоритму ed25519

[illegible]

Figure 4: 4

# Создайте ключи pgp Генерируем ключ

```
Activities Terminal * OS: Fedora 1524 pgp-dell-by -
change %Name, %Comment, %Email or %UserId% after a
any key generation cancel.
pgp-dell-by - S 000 - Full generated key
key (100%) 2.2.18 Copyright (C) 2009 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.

Please select what kind of key you want:
  (1) RSA and ECC (default)
  (2) RSA and ElGamal
  (3) RSA (sign only)
  (4) RSA (sign only)
  (5) Existing key from card
Your selection? 1
Your key must be between 1024 and 4096 bits long.
What keysize do you want? (2048) 4096
Requested keysize is 4096 bits.
Please specify how long the key should be valid.
  0 = key does not expire
  nnn = key expires in n days
  mmm = key expires in m months
  yyy = key expires in y years
key to valid for? (0) 0
key does not expire at all
Is this correct? (Y/n) y

gpg needs to construct a user ID to identify your key.
Real name: Ivan Podgiglyt
Email address: maxcomen@gmail.com
Comment:
You selected this USER ID:
  "Ivan Podgiglyt <maxcomen@gmail.com>"

change %Name, %Comment, %Email or %UserId% after a
you need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
mouse) during the prime generation; this gives the random number
generator a better chance to get enough entropy.
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
mouse) during the prime generation; this gives the random number
generator a better chance to get enough entropy.
any key generated will be automatically created
Use directory "/home/jhi/.gnupg/tmpgpg-revns.d/" created
Use generator default (if it is on "/home/jhi/.gnupg/tmpgpg-revns.d/2024047897974E31E85C7E6D0895AEC8A008.rev")
public and secret key created and signed.

pub rsa4096 2024-09-11 [sc]
    8084F7A970A7897974E31E85C7E6D0895AEC8A008
uid
    Ivan Podgiglyt <maxcomen@gmail.com>
sub rsa4096 2024-09-11 [sc]

gpg> exit 0
```

Figure 5: 5

## Добавление PGP ключа в GitHub

Выводим список ключей и копируем отпечаток приватного ключа

Скопируйте ваш сгенерированный PGP ключ в буфер обмена

[illegible]

Figure 6: 6

# Добавил скопированный ключ в Github аккаунт

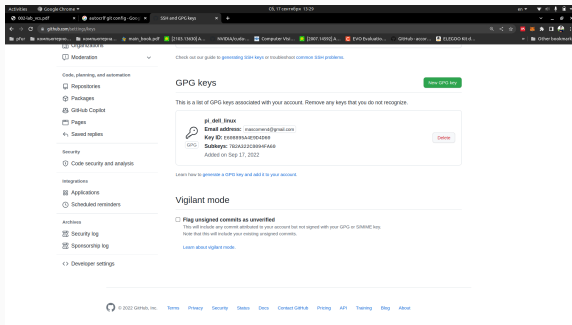


Figure 7: 7

## Настройка автоматических подписей коммитов git

[illegible]

Figure 8: 8

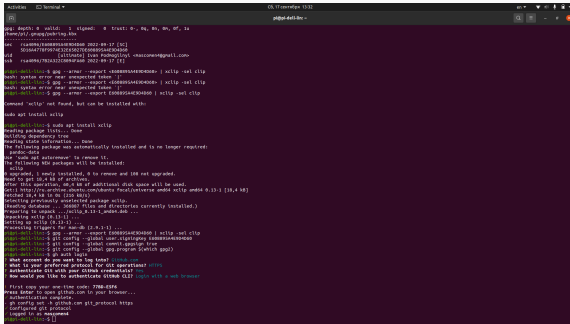


Figure 9: 9

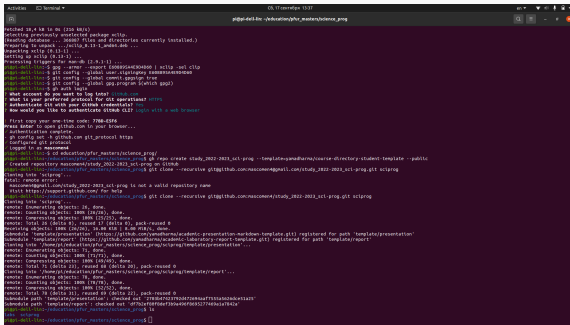


Figure 10: 10

Перейдите в каталог курса:

Удалите лишние файлы. Создайте необходимые каталоги

[illegible]

Figure 11: 11



## Отправьте файлы на сервер

[illegible]

Figure 12: 12

Освоил на практике применение VCS Git, Github.