

Отчёт по лабораторной работе №2

Управление версиями

Тимошенко Анна Михайловна

Содержание

1 Цель работы

Целью данной работы является изучение идеологии и применения средств контроля версий и освоение умений работать с git.

2 Выполнение лабораторной работы

Устанавливаем git, git-flow и gh.

```
antimoshenko@vbox:~$ dnf install git
Для выполнения запрошенной операции требуются привилегии суперпользователя. Пожалуйста, войдите в систему как пользователь с повышенными правами или используйте опции "--assumeno" или "--downloadonly", чтобы выполнить команду без изменения состояния системы.
antimoshenko@vbox:~$ sudo -i
[sudo] пароль для antimoshenko:
root@vbox:~# dnf install git
Обновление и загрузка репозитория:
Fedora 41 - x86_64 - Updates                                100% | 7.2 KiB/s | 23.3 KiB | 00m03s
Fedora 41 - x86_64 - Updates                                100% | 97.4 KiB/s | 2.2 MiB | 00m24s
Репозитории загружены.
Пакет "git-2.48.1-1.fc41.x86_64" уже установлен.

Нечего делать.
root@vbox:~# dnf install gh
Обновление и загрузка репозитория:
Репозитории загружены.
Пакет                                Арх.      Версия      Репозиторий      Размер
Установка:
gh                                   x86_64    2.65.0-1.fc41 updates          42.6 MiB

Сводка транзакции:
Установка:      1 пакета

Общий размер входящих пакетов составляет 10 MiB. Необходимо загрузить 10 MiB.
После этой операции будут использоваться дополнительные 43 MiB (установка 43 MiB, удаление 0 B).
Is this ok [y/N]: y
[1/1] gh-2.65.0-1.fc41.x86_64          70% [===== ] | 63.3 KiB/s | 7.2 MiB | 00m49s
-----
[0/1] Total                            70% [===== ] | 63.3 KiB/s | 7.2 MiB | 00m49s
```

Figure 1: Загрузка пакетов

Зададим имя и email владельца репозитория, кодировку и прочие параметры.

```

root@vbox:~# git config --global user.name "Anna Timoshenko"
root@vbox:~# git config --global user.email "nikol.0715011@gmail.com"
root@vbox:~# git config --global core.quotepath false
root@vbox:~# git config --global init.defaultBranch master
root@vbox:~# git config --global core.autocrlf input
root@vbox:~# git config --global core.safecrlf warn

```

Figure 2: Параметры репозитория

Создаем SSH ключи

```

root@vbox:~# 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 for "/root/.ssh/id_rsa" (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:Wzjk6N11PzCpw6LEb60XPjjGGPqM5RHsEjYsJFXd2Jo root@vbox
The key's randomart image is:
+----[RSA 4096]-----+
|      +               |
|      o o             |
|      o               |
|o. . . E+ . . .      |
| . = o . S o =       |
| o +. + o * o +      |
| ..++=.+.+. . o     |
| .*.o=o*. . . .     |
| ..+.o*oo            |
+----[SHA256]-----+
root@vbox:~#

```

Figure 3: rsa-4096

```

Какой размер ключа Вам необходим? (3072) 4096
Запрошенный размер ключа - 4096 бит
Выберите срок действия ключа.
    0 = не ограничен
    <p> = срок действия ключа - p дней
    <p>w = срок действия ключа - p недель
    <p>m = срок действия ключа - p месяцев
    <p>y = срок действия ключа - p 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: Anna Timoshenko
Адрес электронной почты: nikol.0715011@gmail.com
Примечание:
Вы выбрали следующий идентификатор пользователя:
    "Anna Timoshenko <nikol.0715011@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/F2BF4BF8F732169BE11F8EA833923EA561B24226.rev'.
открытый и секретный ключи созданы и подписаны.

pub  rsa4096 2025-03-04 [SC]
    F2BF4BF8F732169BE11F8EA833923EA561B24226
uid                               Anna Timoshenko <nikol.0715011@gmail.com>
sub   rsa4096 2025-03-04 [E]

```

Figure 4: ed25519

Создаем GPG ключ

```

1GpSHABeY7XSIIYzTuQZYIgy+ZjzmBxawqZxAtih4LREdy4MxeArwZiN0Sk2CCB+
34MIC9MoCu59fwhnAtqKedQtRxCQ2QPAmU3L fus5W5bKFN9BEaxLY9/Ik2w22Yv
8a6ZduBU+N1ScdTBY5QyNHwQP0E9191i4Nnh432KJRMw12pypCDsnITyZuNERCd2
mHoPF078ged/9E3Qqzxu7RXTsc54TmI7L+py33WkTLvoBj4Sg/TpgC2xRWLkK2sg
qBIj0IePn6q1osTa4eBHF916o9G0sR205N5wHyToGH+SxuvegK5JfDqf8nGyDw0D
EIfCJQ7qyYHDXBR3LT5kQgs8wsBe0zTm20G8VX1MvT/zwKaLvItg5y3TLvEuHuv1
Cxa49HID0n4Z/MAXS+dpXBXTYBc6D54VEPwUGZSOYZaLRwvtZ1ogBBGVy2BFT/US
Lm7UDRP7+Gsgw+UBAp/hger40HKAnDiMuJ91kfV+5KaVQ8nG0Nz2b7aqcosPh6t5
RQ/uMnUk2hHCxMcTQz8zGf1jo0Fik0mMdX7kpY7s5sHvMVD4ejKUD+deBKwtK/n9
4wCfavmtLvU+QQY04c5oGDFEweHEXAKuA58Gb6HiZF3B442WRWJoENY1mzyH0zzD
+ttf5TPaUqt2Qrb0dG6juQINBGfG+j8BEAC/TVZnFvSnybGgvuCVk9nE7+61LqG3
4aVTkigEP0ejyPcfnDL0+V/RsSF/mn+gAVcj84SDftcd7zd6St4yxwS3Y5sTXJK
HmPfVvzvZwASRyxStd/S4WpI0aA2zZPjC5qBtUFf32rZxp5cnKsC/XaisYFR89yZ
sDG0JJtDj25J28u2jPa6GXA/pU/Yf5Yqvj9wZNicAsUQA07exngtsivY5hBV+RF
DSMGU3SWUpk+suI3BXfEeaQz5RfdYHeZCSCBNKb44HnJ4iUtoqDo0L8FfzVRr01u
JMyvz7F01dCiVdhcPyeGpzFGu2/HJ0kud6uEkXcEr4KeMv0A1MS63HL5Ka72mVs f
610ag1Jc7N+m9I6xJk210jwcA1Zgk83TF1BRx0453x+jpecDTfwk0a2JCXkASXwM
ID74i3y63UD0GxcrcZCf+mq0M6SxcDdaCUMCTYxK6xHKn8Ujdb4FrwcExqguhIknj
5QwE538cd08rTvuvEuMskPHuRinPN6UYeAcEa8TbZ1fLCp+9c2IbrbPa2kZIV0AE
HPJb1zuxxW4y76+Ap13gvfg0Itr0TxmxA/q02kpnNFyMYxAxm78REm1wF8JGqD4F
0/uLMG1Vsy4BJpUAsx0XgmIKqgVS2EnShK7XCKXyT07CN4q7AR/m0hvc4SxX3jYL
f1nELZcRW5MSGQARAQABiQI2BBgBCAAgFiEE8r9L+PcyFpvhH46oMSI+pWgyQiYF
AmfG+j8CGmwACgkQM5I+pWgyQiZNVxAAoylMXJIPyGhiqsEDeGQ155jjvH+3mU00
nTvPfQq54f8e2f1LEcPcpuWAJcf1NvocISv1MLh3kUaVD0NW/GBCDUwnX4fxY7/F
0TybDvSm6WokqRntJYF2cvbTgLzhe79Nu/zLkfQCN2cG1XgI3Qb2Sh8AVKbaAr/9
EgAQ/cQe7g8CQkR0Uiey5oYKAmf9RFNarVvz1tid0RIknno/7sVpMHIxB5jVDR/X
VByP+bc/fnly9welp4x2TNzW0/1jF2T31h0NpndaAL7Ni98VI1CcX00Dqh5MIYKM
h1IwwGiL41IEYbY+i0cII+Us02/iaVK4Tsfltrzr2LFSrVcrhh7XV92LH8qD2ZhS
UXE8w9CSZPb29FgVBEQe051xm96qMB9agVp0gEcptyimUJZybVMag2JizUytZLAD
dsudbnSnRhrqxGdD277wHfPvtUCRi52Aaw0s0Sn0qDucVx4xp7155MfsWZskef+Z
B3vKTGJ1dn0kPxHmhsMdxA0rnTxco4LIF0YI1q8fVj8pb4TyLV/UnspIUsohxbW
Z1Qgtgo4doDaAHC5Vh0VDwetHteSUCBu3wYnWkE51GFd5fZF1R8pvhH2f19YzI
LKghHqZhw1PueMPZNVFuGNG3envUo500cnB9VJKYJ/BywCsU9nYjHYVg9AkQc3A
qn20Zqth63k=
=HGdN
-----END PGP PUBLIC KEY BLOCK-----
[root@vbox ~]# git config --global user.signingkey 33923EA561B24226
[root@vbox ~]# git config --global commit.gpgsign true
[root@vbox ~]# git config --global gpg.program $(which gpg2)
[root@vbox ~]# git config --global commit.gpgsign true

```

Figure 5: GPG ключ

Добавляем GPG ключ в аккаунт

Настройка автоматических подписей коммитов git

Настройка gh

```
[amtimoshenko@vbox os]$ gh auth login
? Where do you use GitHub? GitHub.com
? What is your preferred protocol for Git operations on this host? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Login with a web browser

! First copy your one-time code: ECCD-B78E
Press Enter to open https://github.com/login/device in your browser... █
```

Figure 6: Связь репозитория с аккаунтом

Загрузка шаблона репозитория и синхронизация

```
[amtimoshenko@vbox os]$ gh repo create study_2024-2025_os-intro --template=yamadharma/course-directory-student-template --public
GraphQL: Could not clone: Name already exists on this account (cloneTemplateRepository)
[amtimoshenko@vbox os]$ git clone --recursive https://github.com/amtimoshenko/study_2024-2025_os-intro os-intro
Клонирование в «os-intro»...
fatal: «https://github.com/amtimoshenko/study_2024-2025_os-intro/» недоступно: Failed to connect to github.com port 443 after 21671 ms: Could not connect to server
[amtimoshenko@vbox os]$ git clone --recursive https://github.com/amtimoshenko/study_2024-2025_os-intro
Клонирование в «study_2024-2025_os-intro»...
remote: Enumerating objects: 36, done.
remote: Counting objects: 100% (36/36), done.
remote: Compressing objects: 100% (35/35), done.
remote: Total 36 (delta 1), reused 21 (delta 0), pack-reused 0 (from 0)
Получение объектов: 100% (36/36), 19.37 КиБ | 461.00 КиБ/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 «/home/amtimoshenko/work/study/2024-2025/os/study_2024-2025_os-intro/template/presentation»...
```

Figure 7: Загрузка шаблона

Подготовка репозитория и коммит изменений

3 Вывод

Мы приобрели практические навыки работы с сервисом github.

4 Контрольные вопросы

1. Что такое системы контроля версий (VCS) и для решения каких задач они предназначаются?

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

2. Объясните следующие понятия VCS и их отношения: хранилище, commit, история, рабочая копия.
 - хранилище - пространство на накопителе где расположен репозиторий
 - commit - сохранение состояния хранилища
 - история - список изменений хранилища (коммитов)
 - рабочая копия - локальная копия сетевого репозитория, в которой работает программист. Текущее состояние файлов проекта, основанное на версии, загруженной из хранилища (обычно на последней)
3. Что представляют собой и чем отличаются централизованные и децентрализованные VCS? Приведите примеры VCS каждого вида.

Централизованные системы контроля версий представляют собой приложения типа клиент-сервер, когда репозиторий проекта существует в единственном экземпляре и хранится на сервере. Доступ к нему осуществлялся через специальное клиентское приложение. В качестве примеров таких программных продуктов можно привести CVS, Subversion.

Распределенные системы контроля версий (Distributed Version Control System, DVCS) позволяют хранить репозиторий (его копию) у каждого разработчика, работающего с данной системой. При этом можно выделить центральный репозиторий (условно), в который будут отправляться изменения из локальных и, с ним же эти локальные репозитории будут синхронизироваться. При работе с такой системой, пользователи периодически синхронизируют свои локальные репозитории с центральным и работают непосредственно со своей локальной копией. После внесения достаточного количества изменений в локальную копию они (изменения) отправляются на сервер. При этом сервер, чаще всего, выбирается условно, т.к. в большинстве DVCS нет такого понятия как “выделенный сервер с центральным репозиторием”.

4. Опишите действия с VCS при единоличной работе с хранилищем.

Один пользователь работает над проектом и по мере необходимости делает коммиты, сохраняя определенные этапы.

5. Опишите порядок работы с общим хранилищем VCS.

Несколько пользователей работают каждый над своей частью проекта. При этом каждый должен работать в своей ветки. При завершении работы ветка пользователя сливается с основной веткой проекта.

6. Каковы основные задачи, решаемые инструментальным средством git?
 - Ведение истории версий проекта: журнал (log), метки (tags), ветвления (branches).
 - Работа с изменениями: выявление (diff), слияние (patch, merge).
 - Обеспечение совместной работы: получение версии с сервера, загрузка обновлений на сервер.
7. Назовите и дайте краткую характеристику командам git.
 - git config - установка параметров
 - git status - полный список изменений файлов, ожидающих коммита
 - git add . - сделать все измененные файлы готовыми для коммита.
 - git commit -m "[descriptive message]" - записать изменения с заданным сообщением.
 - git branch - список всех локальных веток в текущей директории.
 - git checkout [branch-name] - переключиться на указанную ветку и обновить рабочую директорию.
 - git merge [branch] — соединить изменения в текущей ветке с изменениями из заданной.
 - git push - запустить текущую ветку в удаленную ветку.
 - git pull - загрузить историю и изменения удаленной ветки и произвести слияние с текущей веткой.
8. Приведите примеры использования при работе с локальным и удалённым репозиториями.
 - git remote add [имя] [url] — добавляет удалённый репозиторий с заданным именем;
 - git remote remove [имя] — удаляет удалённый репозиторий с заданным именем;
 - git remote rename [старое имя] [новое имя] — переименовывает удалённый репозиторий;
 - git remote set-url [имя] [url] — присваивает репозиторию с именем новый адрес;
 - git remote show [имя] — показывает информацию о репозитории.
9. Что такое и зачем могут быть нужны ветви (branches)?

Ветвление — это возможность работать над разными версиями проекта: вместо одного списка с упорядоченными коммитами история будет расходиться в определённых точках. Каждая ветвь содержит легковесный указатель HEAD на последний коммит, что позволяет без лишних затрат создать много веток. Ветка по умолчанию называется master, но лучше назвать её в соответствии с разрабатываемой в ней функциональностью.

10. Как и зачем можно игнорировать некоторые файлы при commit?

Зачастую нам не нужно, чтобы Git отслеживал все файлы в репозитории, потому что в их число могут входить: