

Отчёт по лабораторной работе №6

Дисциплина : операционная система

Замула Егор Сергеевич

Содержание

Цель работы	5
Задание	6
Теоретическое введение	7
Выполнение лабораторной работы	9
Контрольные вопросы	14
Выводы	15
Список литературы	16

Список иллюстраций

1	Установка git	9
2	Настройка git	9
3	Создание ssh ключа	10
4	Добавление SSH ключа	10
5	pgr ключ	11
6	Добавляем pgr ключ	11
7	sec	12
8	Подписи	12
9	Клонируем	12
10	Настройка каталога	13

Список таблиц

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.
- Зарегистрироваться на Github.
- Создать локальный каталог для выполнения заданий по предмету.

Теоретическое введение

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.

В классических системах контроля версий используется централизованная модель, предполагающая наличие единого репозитория для хранения файлов. Выполнение большинства функций по управлению версиями осуществляется специальным сервером. Участник проекта (пользователь) перед началом работы посредством определённых команд получает нужную ему версию файлов. После внесения изменений, пользователь размещает новую версию в хранилище. При этом предыдущие версии не удаляются из центрального хранилища и к ним можно вернуться в любой момент. Сервер может сохранять не полную версию изменённых файлов, а производить так называемую дельта-компрессию — сохранять только изменения между последовательными версиями, что позволяет уменьшить объём хранимых данных.

Системы контроля версий поддерживают возможность отслеживания и разрешения конфликтов, которые могут возникнуть при работе нескольких человек над одним файлом. Можно объединить (слить) изменения, сделанные разными участниками (автоматически или вручную), вручную выбрать нужную версию,

отменить изменения вовсе или заблокировать файлы для изменения. В зависимости от настроек блокировка не позволяет другим пользователям получить рабочую копию или препятствует изменению рабочей копии файла средствами файловой системы ОС, обеспечивая таким образом, привилегированный доступ только одному пользователю, работающему с файлом.

Системы контроля версий также могут обеспечивать дополнительные, более гибкие функциональные возможности. Например, они могут поддерживать работу с несколькими версиями одного файла, сохраняя общую историю изменений до точки ветвления версий и собственные истории изменений каждой ветви. Кроме того, обычно доступна информация о том, кто из участников, когда и какие изменения вносил. Обычно такого рода информация хранится в журнале изменений, доступ к которому можно ограничить.

В отличие от классических, в распределённых системах контроля версий центральный репозиторий не является обязательным.

Среди классических VCS наиболее известны CVS, Subversion, а среди распределённых — Git, Bazaar, Mercurial. Принципы их работы схожи, отличаются они в основном синтаксисом используемых в работе команд.

Выполнение лабораторной работы

1. Установка системы git в нашу ОС

```
eszamula@fedora:~$ dnf install git
Ошибка: Эту команду нужно запускать с привилегиями суперпользователя (на большинстве систем - под именем пользователя root).
eszamula@fedora:~$ sudo -i
[sudo] пароль для eszamula:
root@fedora:~# dnf install git
Fedora 39 - x86_64 - Updates
Fedora 39 - x86_64 - Updates
Последняя проверка окончания срока действия метаданных: 0:00:09 назад, Сб 24 фев 2024 11:46:26.
Пакет git-2.43.2-1.fc39.x86_64 уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
root@fedora:~#
```

Рис. 1: Установка git

2. Проводим базовую настройку git

```
eszamula@fedora:~$ git config --global user.name "egorzam21"
eszamula@fedora:~$ git config --global user.email "1132230796@pfur.ru"
eszamula@fedora:~$ git config --global core.quotepath false
eszamula@fedora:~$ git config --global init.defaultBranch master
eszamula@fedora:~$ git config --global core.autocrlf input
eszamula@fedora:~$ git config --global core.safecrlf warn
eszamula@fedora:~$
```

Рис. 2: Настройка git

3. Создаем SSH ключа по алгоритму rsa с ключём размером 4096 бит

```

eszamula@fedora:~$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/eszamula/.ssh/id_rsa):
Created directory '/home/eszamula/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/eszamula/.ssh/id_rsa
Your public key has been saved in /home/eszamula/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:4MKUHKI8kV8aNRQzGYoy5ookTm64wTyc0INH9zi7fIM eszamula@fedora
The key's randomart image is:
+---[RSA 4096]-----+
| .o +0+          |
|.oo+.,.+         |
|+=o.O .          |
|+=.* + .         |
|+++ = o S        |
|#... +          |
|=X ..           |
|.O..E.O          |
|. o. .           |
+---[SHA256]-----+
eszamula@fedora:~$

```

Рис. 3: Создание ssh ключа

4. Добавляем SSH ключ на гитхаб в разделе settings

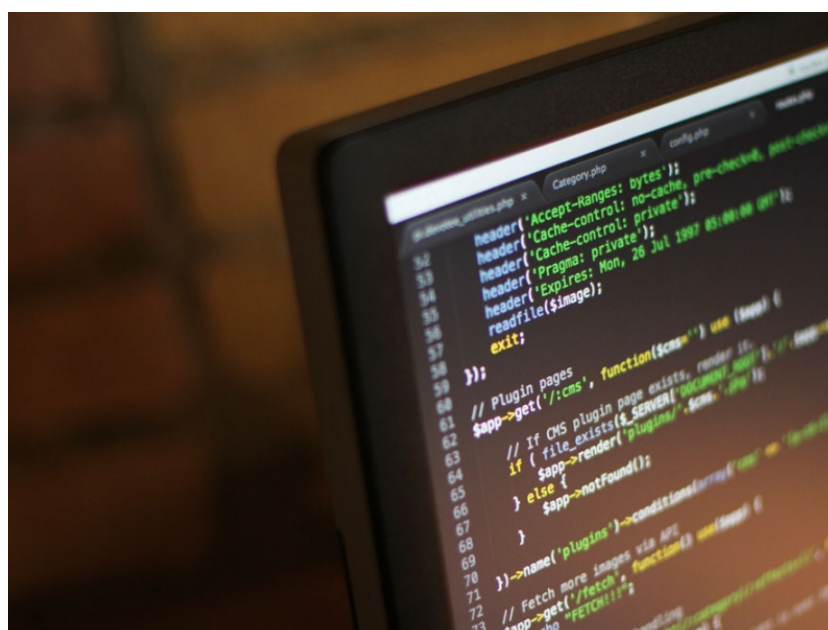


Рис. 4: Добавление SSH ключа

5. Создание pgr ключа

```
eszmula@fedora:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/eszmula/.ssh/id_rsa):
Created new keys.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/eszmula/.ssh/id_rsa.
Your public key has been saved in /home/eszmula/.ssh/id_rsa.pub.
The key fingerprint is:
a99f870583418275f64985f4593a3e69864a0666 eszmula@fedora:~$ cat /home/eszmula/.ssh/id_rsa.pub
rsa4096 2024-02-24 [SC]
A99F870583418275F64985F4593A3E69864A0666
uid
eszmula@fedora:~$
```

Рис. 5: ргр ключ

6. Добавляем ключ на гитхаб (точно так же как и SSH)

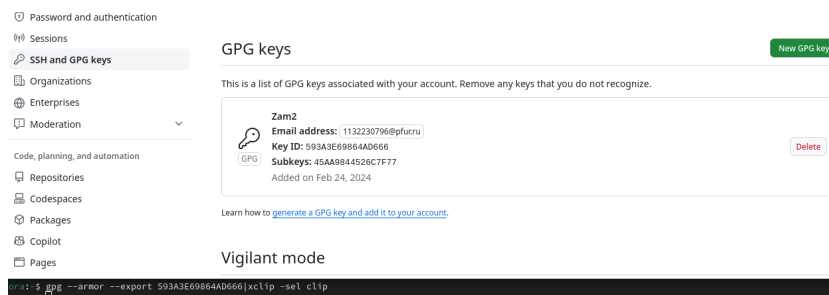


Рис. 6: Добавляем ргр ключ

7. Получение ses параметра

```

eszamula@fedora:~$ gpg --list-secret-keys --keyid-format LONG
gpg: проверка таблицы доверия
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: глубина: 0 достоверных: 1 подписанных: 0 доверие: 0-, 0q, 0n, 0m, 0f, 1u
[keyboard]
-----
sec   rsa4096/593A3E69864AD666 2024-02-24 [SC]
      A99F870583418275F64985F4593A3E69864AD666
uid   [ абсолютно ] egorzam21 <1132230796@pfur.ru>
ssb   rsa4096/45AA9844526C7F77 2024-02-24 [E]

eszamula@fedora:~$ gpg --armor --export <PGP Fingerprint> | xclip -sel clip
bash: синтаксическая ошибка рядом с неожиданным маркером «|»
eszamula@fedora:~$ gpg --armor --export <PGP Fingerprint>|xclip -sel clip
bash: PGP: Нет такого файла или каталога
eszamula@fedora:~$ gpg --armor --export <593A3E69864AD666>|xclip -sel clip
bash: 593A3E69864AD666: Нет такого файла или каталога
eszamula@fedora:~$ gpg --armor --export 593A3E69864AD666|xclip -sel clip
eszamula@fedora:~$

```

Рис. 7: sec

8. Настройка автоматических подписей коммитов git

```

eszamula@fedora:~$ git config --global user.signingkey 593A3E69864AD666
eszamula@fedora:~$ git config --global commit.gpgsign true
eszamula@fedora:~$ git config --global gpg.program $(which gpg2)
eszamula@fedora:~$

```

Рис. 8: Подписи

9. Клонировем репозиторий

```

eszamula@fedora:~$ git clone --recursive git@github.com:egor2am21/study_2022-2023_os-intro.git os-intro
Клонирование в «os-intro»...
The authenticity of host 'github.com (140.82.121.4)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvV6TujJhbpZisF/zLDA0zPMSvHdK4UvCoQU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
ERROR: Repository not found.
fatal: Не удалось прочитать из внешнего репозитория.

Удостоверьтесь, что у вас есть необходимые права доступа
и репозиторий существует.
eszamula@fedora:~$

```

Рис. 9: Клонировем

10. Настройка каталога курса

```
Created repository egorzam21/study_2023-2024_os-intro on GitHub
https://github.com/egorzam21/study_2023-2024_os-intro
esizamula@fedora:~/work/study/2022-2023/Операционные системы$ git clone --recursive git@github.com:egorzam21/study_2022-2023_os-intro.git os-intro
Клонирование в «os-intro»...
ERROR: Repository not found.
Fatal: Не удалось прочитать из внешнего репозитория.

Удостоверьтесь, что у вас есть необходимые права доступа
и репозиторий существует.
esizamula@fedora:~/work/study/2022-2023/Операционные системы$ mkdir -p ~/work/study/2022-2023/Операционные системы/os-intro
esizamula@fedora:~/work/study/2022-2023/Операционные системы/os-intro$ rm package.json
rm: невозможно удалить 'package.json': Нет такого файла или каталога
esizamula@fedora:~/work/study/2022-2023/Операционные системы/os-intro$ git clone --recursive git@github.com:egorzam21/study_2022-2023_os-intro.git os-intro
Клонирование в «os-intro»...
ERROR: Repository not found.
Fatal: Не удалось прочитать из внешнего репозитория.

Удостоверьтесь, что у вас есть необходимые права доступа
и репозиторий существует.
esizamula@fedora:~/work/study/2022-2023/Операционные системы/os-intro$ git clone --recursive git@github.com:egorzam21/study_2023-2024_os-intro.git os-intro
Клонирование в «os-intro»...
remote: Enumerating objects: 32, done.
remote: Counting objects: 100% (32/32), done.
remote: Compressing objects: 100% (31/31), done.
remote: Total 32 (delta 1), reused 18 (delta 0), pack-reused 0
Получение объектов: 100% (32/32), 18.60 Киб | 1.86 Мб/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/esizamula/work/study/2022-2023/Операционные системы/os-intro/os-intro/template/presentation»...
remote: Enumerating objects: 95, done.
remote: Counting objects: 100% (95/95), done.
remote: Compressing objects: 100% (67/67), done.
remote: Total 95 (delta 34), reused 97 (delta 26), pack-reused 0
Получение объектов: 100% (95/95), 96.99 Киб | 1.37 Мб/с, готово.
Определение изменений: 100% (34/34), готово.
Клонирование в «/home/esizamula/work/study/2022-2023/Операционные системы/os-intro/os-intro/template/report»...
remote: Enumerating objects: 126, done.
remote: Counting objects: 100% (126/126), done.
remote: Compressing objects: 100% (87/87), done.
remote: Total 126 (delta 52), reused 108 (delta 34), pack-reused 0
Получение объектов: 100% (126/126), 335.80 Киб | 2.54 Мб/с, готово.
Определение изменений: 100% (52/52), готово.
Submodule path 'template/presentation': checked out '4b01761813e107d09e6848ff1c72c60a304f24c'
Submodule path 'template/report': checked out '7c31ab8e5dfaf8db2d67caeb8a19ef8928ced8e'
esizamula@fedora:~/work/study/2022-2023/Операционные системы/os-intro$
```

Рис. 10: Настройка каталога

```
remote: Enumerating objects: 95, done.
remote: Counting objects: 100% (95/95), done.
remote: Compressing objects: 100% (67/67), done.
remote: Total 95 (delta 34), reused 97 (delta 26), pack-reused 0
Получение объектов: 100% (95/95), 96.99 Киб | 1.37 Мб/с, готово.
Определение изменений: 100% (34/34), готово.
Клонирование в «/home/esizamula/work/study/2022-2023/Операционные системы/os-intro/os-intro/template/presentation»...
remote: Enumerating objects: 126, done.
remote: Counting objects: 100% (126/126), done.
remote: Compressing objects: 100% (87/87), done.
remote: Total 126 (delta 52), reused 108 (delta 34), pack-reused 0
Получение объектов: 100% (126/126), 335.80 Киб | 2.54 Мб/с, готово.
Определение изменений: 100% (52/52), готово.
Submodule path 'template/presentation': checked out '4b01761813e107d09e6848ff1c72c60a304f24c'
Submodule path 'template/report': checked out '7c31ab8e5dfaf8db2d67caeb8a19ef8928ced8e'
esizamula@fedora:~/work/study/2022-2023/Операционные системы/os-intro$ rm package.json
rm: невозможно удалить 'package.json': Нет такого файла или каталога
esizamula@fedora:~/work/study/2022-2023/Операционные системы/os-intro$ cd -/work/study/2022-2023/Операционные системы/os-intro/os-intro
esizamula@fedora:~/work/study/2022-2023/Операционные системы/os-intro/os-intro$ delete mode 100644 package.json
esizamula@fedora:~/work/study/2022-2023/Операционные системы/os-intro/os-intro$
Usage:
  make <target>

Targets:
  list              List of courses
  prepare           Generate directories structure
  submodule         Update submodules

esizamula@fedora:~/work/study/2022-2023/Операционные системы/os-intro/os-intro$
esizamula@fedora:~/work/study/2022-2023/Операционные системы/os-intro/os-intro$ [master 3dd3ca4] feat(main): make course structure
2 files changed, 1 insertion(+), 14 deletions(-)
delete mode 100644 package.json
esizamula@fedora:~/work/study/2022-2023/Операционные системы/os-intro/os-intro$
Перечисление объектов: 5, готово.
Подсчет объектов: 100% (5/5), готово.
При скачивании изменений используется до 3 потоков
Сжатие объектов: 100% (2/2), готово.
Запись объектов: 100% (3/3), 949.60 Киб/с, готово.
Всего 3 (изменений 1), повторно использовано 0 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано 0 (изменений 0).
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:egorzam21/study_2023-2024_os-intro.git
3071314..3dd3ca4 master -> master
esizamula@fedora:~/work/study/2022-2023/Операционные системы/os-intro/os-intro$
```

11. Отправка на сервер нашего отредактированного репозитория

Контрольные вопросы

Что такое системы контроля версий (VCS) и для решения каких задач они предназначены?

Объясните следующие понятия VCS и их отношения: хранилище, commit, история, рабочий каталог.

Что представляют собой и чем отличаются централизованные и децентрализованные VCS?

Опишите действия с VCS при единоличной работе с хранилищем.

Опишите порядок работы с общим хранилищем VCS.

Каковы основные задачи, решаемые инструментальным средством git?

Назовите и дайте краткую характеристику командам git.

Приведите примеры использования при работе с локальным и удалённым репозиториями.

Что такое и зачем могут быть нужны ветви (branches)?

Как и зачем можно игнорировать некоторые файлы при commit?

Выводы

В ходе выполнения работы изучил git

Список литературы