

# **Отчет по лабораторной работе №1**

**Информационная безопасность**

Чекалова Лилия Руслановна

# Содержание

<b>Цель работы</b>	<b>5</b>
<b>Задание</b>	<b>6</b>
<b>Теоретическое введение</b>	<b>7</b>
<b>Выполнение лабораторной работы</b>	<b>8</b>
<b>Выводы</b>	<b>22</b>
<b>Список литературы</b>	<b>23</b>

## **Список таблиц**

## Список иллюстраций

0.1	Базовая настройка git . . . . .	8
0.2	Генерация ssh-ключей . . . . .	9
0.3	Генерация grpg-ключа, 1 . . . . .	10
0.4	Генерация grpg-ключа, 2 . . . . .	11
0.5	Копирование grpg-ключа . . . . .	11
0.6	Подключение grpg-ключа . . . . .	12
0.7	Копирование репозитория . . . . .	12
0.8	Создание каталога по шаблону . . . . .	12
0.9	Внесение изменений на сервер, 1 . . . . .	13
0.10	Внесение изменений на сервер, 2 . . . . .	13
0.11	Имя и тип ОС . . . . .	14
0.12	Объем оперативной памяти . . . . .	14
0.13	Объем жесткого диска . . . . .	15
0.14	Подключение образа оптического диска . . . . .	15
0.15	Выбор программ . . . . .	16
0.16	Настройка имени узла . . . . .	17
0.17	Задание пароля для root . . . . .	18
0.18	Обзор установки . . . . .	19
0.19	Подключение образа диска дополнений гостевой ОС . . . . .	20
0.20	Поиск информации с помощью dmesg, 1 . . . . .	20
0.21	Поиск информации с помощью dmesg, 2 . . . . .	21

## Цель работы

- Приобретение практических навыков установки операционной системы на виртуальную машину
- Освоение умений по работе с git
- Составление унифицированной структуры рабочего каталога лабораторных работ

## Задание

- Генерация ssh- и gpg-ключей
- Создание репозитория по шаблону
- Установка операционной системы с указанными характеристиками на виртуальную машину
- Поиск информации о системе

# 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.

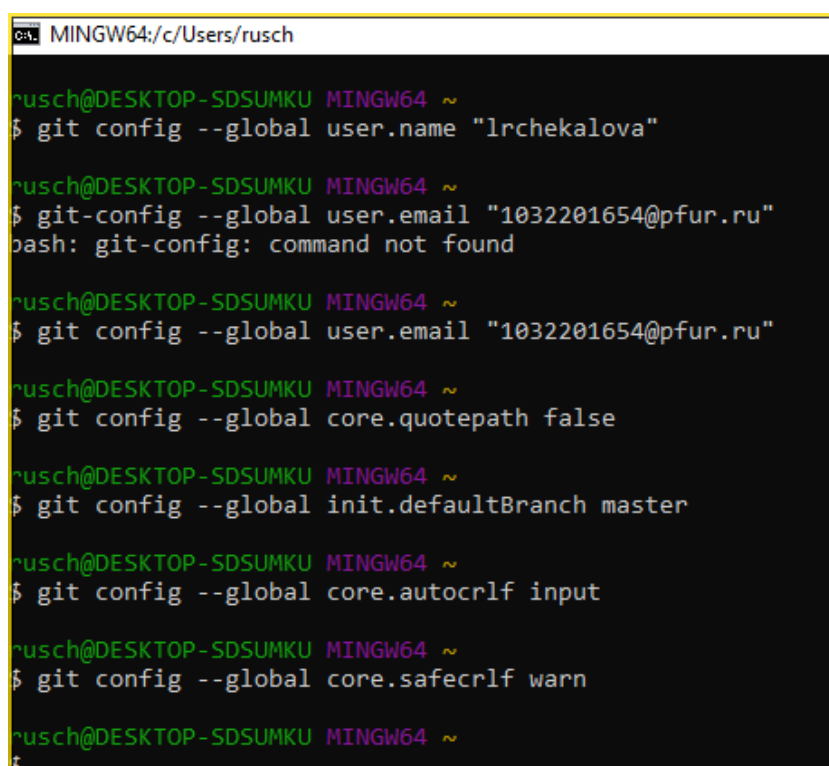
При внесении изменений в содержание проекта система контроля версий позволяет их фиксировать, совмещать изменения, произведённые разными участниками проекта, производить откат к любой более ранней версии проекта, если это требуется.

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.

Более подробно о см. в [1,2].

# Выполнение лабораторной работы

Для выполнения лабораторной работы сначала была произведена настройка параметров git, таких как имя и почта пользователя, установка кодирования и другие (рис. 0.1).



```
MINGW64:/c/Users/rusch
rusch@DESKTOP-SDSUMKU MINGW64 ~
$ git config --global user.name "lrchekalova"

rusch@DESKTOP-SDSUMKU MINGW64 ~
$ git-config --global user.email "1032201654@pfur.ru"
bash: git-config: command not found

rusch@DESKTOP-SDSUMKU MINGW64 ~
$ git config --global user.email "1032201654@pfur.ru"

rusch@DESKTOP-SDSUMKU MINGW64 ~
$ git config --global core.quotepath false

rusch@DESKTOP-SDSUMKU MINGW64 ~
$ git config --global init.defaultBranch master

rusch@DESKTOP-SDSUMKU MINGW64 ~
$ git config --global core.autocrlf input

rusch@DESKTOP-SDSUMKU MINGW64 ~
$ git config --global core.safecrlf warn

rusch@DESKTOP-SDSUMKU MINGW64 ~
$
```

Рис. 0.1: Базовая настройка git

Далее были сгенерированы ssh-ключи по алгоритмам rsa и ed25519 (рис. 0.2).



```
MINGW64:/c/Users/rusch
$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/rusch/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/rusch/.ssh/id_rsa
Your public key has been saved in /c/Users/rusch/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:JwHaGG4+HIj9XfzSlKGlbYqfmJ/qo+o34+d183mxcek rusch@DESKTOP-SDSUMKU
The key's randomart image is:
+---[RSA 4096]-----+
  . . o
  o o = o + o
  . o * . * o
  = o . *
  = . S *
  . * o..
  . o o . =
  + o* + o .oE
  .o+oB*+= o.
+-----[SHA256]-----+

rusch@DESKTOP-SDSUMKU MINGW64 ~
$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/c/Users/rusch/.ssh/id_ed25519):
/c/Users/rusch/.ssh/id_ed25519 already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/rusch/.ssh/id_ed25519
Your public key has been saved in /c/Users/rusch/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:ihEv0tSOyM21SXyN2GDdb2g64ZZf2pmZd1yqHIwHc7Y rusch@DESKTOP-SDSUMKU
The key's randomart image is:
+---[ED25519 256]---+
  .+
  +.= o
  o *o+ .
  . *.Bo+o
  +.B=.S o
  .*+ ..B .
  ..O.+.=E. o
  o *o..+
  .oo
+-----[SHA256]-----+
```

Рис. 0.2: Генерация ssh-ключей

Также был сгенерирован gpg-ключ с указанными опциями (рис. 0.3-0.4).

```

C:\MINGW64\c\Users\rusch
rusch@DESKTOP-SDSUMKU MINGW64 ~
$ gpg --full-generate-key
gpg (GnuPG) 2.2.29-unknown; Copyright (C) 2021 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.

gpg: directory '/c/Users/rusch/.gnupg' created
gpg: keybox '/c/Users/rusch/.gnupg/pubring.kbx' created
Please select what kind of key you want:
  (1) RSA and RSA (default)
  (2) DSA and Elgamal
  (3) DSA (sign only)
  (4) RSA (sign only)
  (14) Existing key from card
Your selection? 1
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (3072) 4096
Requested keysize is 4096 bits
Please specify how long the key should be valid.
  0 = key does not expire
  <n> = key expires in n days
  <n>w = key expires in n weeks
  <n>m = key expires in n months
  <n>y = key expires in n years
Key is valid for? (0) 0
Key does not expire at all
Is this correct? (y/N) y

GnuPG needs to construct a user ID to identify your key.

Real name: lrcchekalova
Email address: 1032201654@pfur.ru
Comment:
You selected this USER-ID:
  "lrcchekalova <1032201654@pfur.ru>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? o
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
disks) during the prime generation; this gives the random number
generator a better chance to gain 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
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
gpg: /c/Users/rusch/.gnupg/trustdb.gpg: trustdb created

```

Рис. 0.3: Генерация gpg-ключа, 1

```
MINGW64/c/Users/rusch
Real name: lrchekalova
Email address: 1032201654@pfur.ru
Comment:
You selected this USER-ID:
"lrchekalova <1032201654@pfur.ru>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? o
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
disks) during the prime generation; this gives the random number
generator a better chance to gain 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
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
gpg: /c/Users/rusch/.gnupg/trustdb.gpg: trustdb created
gpg: key 85727667F1B8AC75 marked as ultimately trusted
gpg: directory '/c/Users/rusch/.gnupg/openpgp-revocs.d' created
gpg: revocation certificate stored as '/c/Users/rusch/.gnupg/openpgp-revocs.d/8F24261189430C97D0D1E01C85727667F1B8AC75.rev'
public and secret key created and signed.

pub   rsa4096 2023-09-07 [SC]
      8F24261189430C97D0D1E01C85727667F1B8AC75
uid           [ultimate] lrchekalova <1032201654@pfur.ru>
sub   rsa4096 2023-09-07 [E]

rusch@DESKTOP-SDSUMKU MINGW64 ~
$
```

Рис. 0.4: Генерация gpg-ключа, 2

Содержимое ключа было скопировано (рис. 0.5) и добавлено к аккаунту на сайте GitHub (рис. 0.6).

```
rusch@DESKTOP-SDSUMKU MINGW64 ~
$ gpg --list-secret-keys --keyid-format LONG
gpg: checking the trustdb
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
/c/Users/rusch/.gnupg/pubring.kbx
-----
sec   rsa4096/85727667F1B8AC75 2023-09-07 [SC]
      8F24261189430C97D0D1E01C85727667F1B8AC75
uid           [ultimate] lrchekalova <1032201654@pfur.ru>
ssb   rsa4096/5010AACDB1C8C67 2023-09-07 [E]

rusch@DESKTOP-SDSUMKU MINGW64 ~
$ gpg --armor --export 85727667F1B8AC75 | xclip -sel clip
bash: xclip: command not found
gpg: [stdout]: write error: Broken pipe
gpg: filter_flush failed on close: Broken pipe

rusch@DESKTOP-SDSUMKU MINGW64 ~
$ gpg --armor --export 85727667F1B8AC75 | clip
```

Рис. 0.5: Копирование gpg-ключа

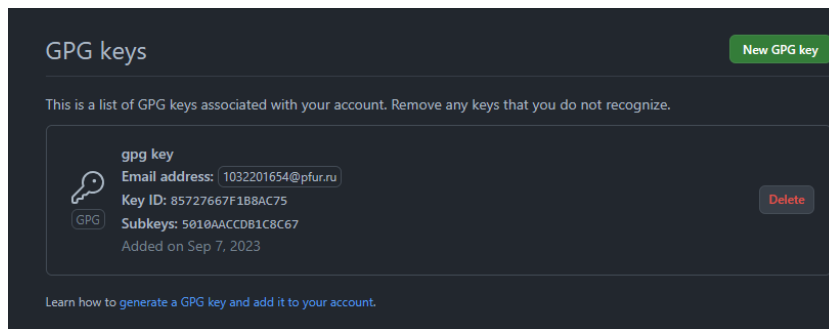


Рис. 0.6: Подключение grg-ключа

Далее был скопирован репозиторий с помощью команды `git clone` (рис. 0.7) и с помощью команды `make COURSE` был построен рабочий каталог по шаблону (рис. 0.8).

```

pushch@DESKTOP-SDSUMKU MINGW64 ~/work/study/2023-2024/Информационная безопасность
$ git clone --recursive git@github.com:lrchiekalova/study_2023-2024_infosec.git infosec
Cloning into 'infosec'...
remote: Enumerating objects: 28, done.
remote: Counting objects: 100% (28/28), done.
remote: Compressing objects: 100% (27/27), done.
Receiving objects: 35% (10/28) 11 (delta 0), pack-reused 0
Receiving objects: 100% (28/28), 17.44 KiB | 2.18 MiB/s, done.
Resolving deltas: 100% (1/1), done.
Submodule 'template/presentation' (https://github.com/yamadharma/academic-presentation-markdown-template.git) registered for path 'template/presentation'
Submodule 'template/report' (https://github.com/yamadharma/academic-laboratory-report-template.git) registered for path 'template/report'
Cloning into 'C:/Users/rusch/work/study/2023-2024/Информационная безопасность/infosec/template/presentation'...
remote: Enumerating objects: 82, done.
remote: Counting objects: 100% (82/82), done.
remote: Compressing objects: 100% (57/57), done.
remote: Total 82 (delta 28), reused 77 (delta 23), pack-reused 0
Receiving objects: 100% (82/82), 92.90 KiB | 1.04 MiB/s, done.
Resolving deltas: 100% (28/28), done.
Cloning into 'C:/Users/rusch/work/study/2023-2024/Информационная безопасность/infosec/template/report'...
remote: Enumerating objects: 181, done.
remote: Counting objects: 100% (181/181), done.
remote: Compressing objects: 100% (78/78), done.
remote: Total 181 (delta 48), reused 88 (delta 27), pack-reused 0
Receiving objects: 100% (181/181), 327.25 KiB | 1.63 MiB/s, done.
Resolving deltas: 100% (40/40), done.
Submodule path 'template/presentation': checked out 'b1be380ee91f5089264cb755d316174540b753e'
Submodule path 'template/report': checked out '1d1b61dcac9c287a83917b2e3aeef1a33b1e302'
pushch@DESKTOP-SDSUMKU MINGW64 ~/work/study/2023-2024/Информационная безопасность
$

```

Рис. 0.7: Копирование репозитория

```

pushch@DESKTOP-SDSUMKU MINGW64 ~/work/study/2023-2024/Информационная безопасность/infosec (master)
$ make COURSE=infosec

```

Рис. 0.8: Создание каталога по шаблону

Изменения были отправлены на сервер (рис. 0.9-0.10).

```

rusch@DESKTOP-SDSUMKU MINGW64 ~/work/study/2023-2024/Информационная безопасность/infosec (master)
$ git add .

rusch@DESKTOP-SDSUMKU MINGW64 ~/work/study/2023-2024/Информационная безопасность/infosec (master)
$ git commit -am 'feat(main): make course structure'
[master 5593670] feat(main): make course structure
150 files changed, 41044 insertions(+), 14 deletions(-)
create mode 100644 labs/README.md
create mode 100644 labs/README.ru.md
create mode 100644 labs/lab1/presentation/Makefile
create mode 100644 labs/lab1/presentation/image/kulyabov.jpg
create mode 100644 labs/lab1/presentation/presentation.md
create mode 100644 labs/lab1/report/Makefile
create mode 100644 labs/lab1/report/bib/cite.bib
create mode 100644 labs/lab1/report/image/placeimg_800_600_tech.jpg
create mode 100644 labs/lab1/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100644 labs/lab1/report/pandoc/filters/pandoc_eqnos.py
create mode 100644 labs/lab1/report/pandoc/filters/pandoc_fignos.py
create mode 100644 labs/lab1/report/pandoc/filters/pandoc_secnos.py
create mode 100644 labs/lab1/report/pandoc/filters/pandoc_tablenos.py
create mode 100644 labs/lab1/report/pandoc/filters/pandocxnos/_init_.py
create mode 100644 labs/lab1/report/pandoc/filters/pandocxnos/core.py
create mode 100644 labs/lab1/report/pandoc/filters/pandocxnos/main.py
create mode 100644 labs/lab1/report/pandoc/filters/pandocxnos/pandocattributes.py
create mode 100644 labs/lab1/report/report.md
create mode 100644 labs/lab2/presentation/Makefile
create mode 100644 labs/lab2/presentation/image/kulyabov.jpg
create mode 100644 labs/lab2/presentation/presentation.md
create mode 100644 labs/lab2/report/Makefile
create mode 100644 labs/lab2/report/bib/cite.bib
create mode 100644 labs/lab2/report/image/placeimg_800_600_tech.jpg
create mode 100644 labs/lab2/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100644 labs/lab2/report/pandoc/filters/pandoc_eqnos.py
create mode 100644 labs/lab2/report/pandoc/filters/pandoc_fignos.py
create mode 100644 labs/lab2/report/pandoc/filters/pandoc_secnos.py
create mode 100644 labs/lab2/report/pandoc/filters/pandoc_tablenos.py
create mode 100644 labs/lab2/report/pandoc/filters/pandocxnos/_init_.py
create mode 100644 labs/lab2/report/pandoc/filters/pandocxnos/core.py
create mode 100644 labs/lab2/report/pandoc/filters/pandocxnos/main.py
create mode 100644 labs/lab2/report/pandoc/filters/pandocxnos/pandocattributes.py
create mode 100644 labs/lab2/report/report.md
create mode 100644 labs/lab3/presentation/Makefile

```

Рис. 0.9: Внесение изменений на сервер, 1

```

MINGW64/c/Users/rusch/work/study/2023-2024/Информационная безопасность/infosec
create mode 100644 presentation/presentation/presentation.md
create mode 100644 presentation/report/Makefile
create mode 100644 presentation/report/bib/cite.bib
create mode 100644 presentation/report/image/placeimg_800_600_tech.jpg
create mode 100644 presentation/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100644 presentation/report/pandoc/filters/pandoc_eqnos.py
create mode 100644 presentation/report/pandoc/filters/pandoc_fignos.py
create mode 100644 presentation/report/pandoc/filters/pandoc_secnos.py
create mode 100644 presentation/report/pandoc/filters/pandoc_tablenos.py
create mode 100644 presentation/report/pandoc/filters/pandocxnos/_init_.py
create mode 100644 presentation/report/pandoc/filters/pandocxnos/core.py
create mode 100644 presentation/report/pandoc/filters/pandocxnos/main.py
create mode 100644 presentation/report/pandoc/filters/pandocxnos/pandocattributes.py
create mode 100644 presentation/report/report.md

rusch@DESKTOP-SDSUMKU MINGW64 ~/work/study/2023-2024/Информационная безопасность/infosec (master)
$ git push
Enumerating objects: 35, done.
Counting objects: 100% (35/35), done.
Delta compression using up to 8 threads
Compressing objects: 100% (29/29), done.
Writing objects: 100% (34/34), 342.73 KiB | 484.00 KiB/s, done.
Total 34 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 1 local object.
To github.com:lrchekalova/study_2023-2024_infosec.git
   ad8a8f8..5593670  master -> master

```

Рис. 0.10: Внесение изменений на сервер, 2

После этого я перешла к установке операционной системы на виртуальную машину. Сначала были заданы имя и тип операционной системы (рис. 0.11).

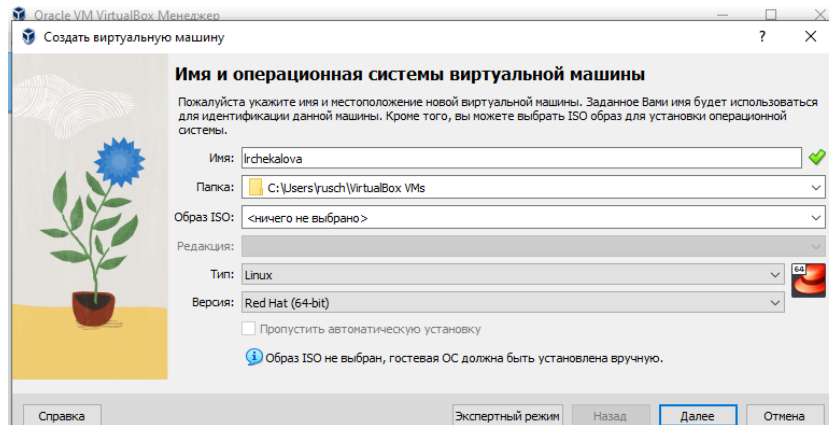


Рис. 0.11: Имя и тип ОС

Были установлены объемы оперативной памяти (рис. 0.12) и жесткого диска (рис. 0.13).

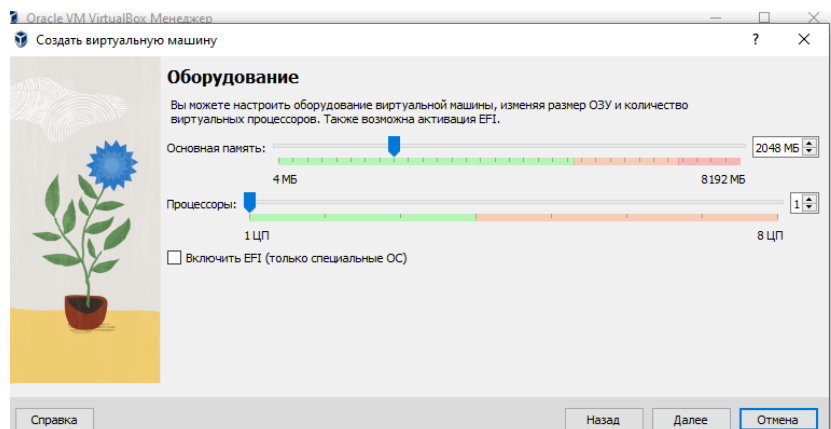


Рис. 0.12: Объем оперативной памяти

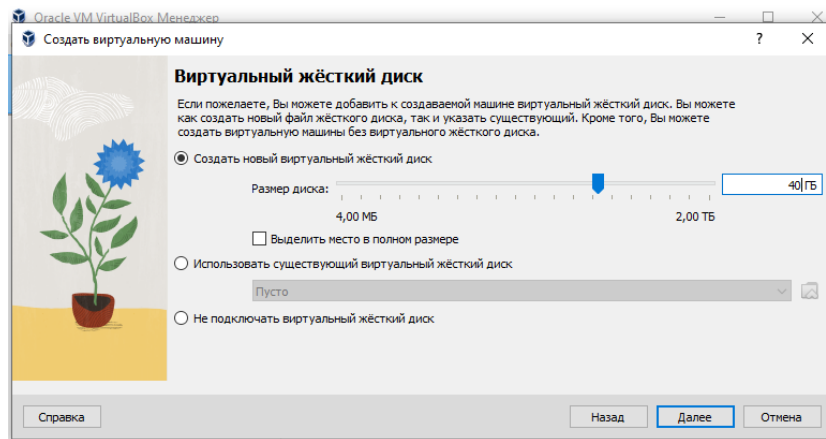


Рис. 0.13: Объем жесткого диска

Был подключен образ оптического диска через окно “Носители” виртуальной машины (рис. 0.14).

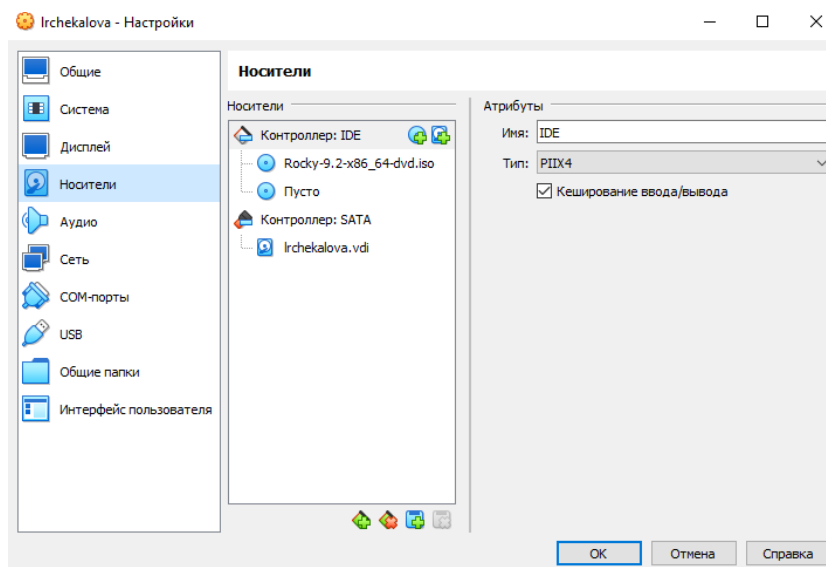


Рис. 0.14: Подключение образа оптического диска

Указали в разделе выбора программ сервер с GUI и средства разработки (рис. 0.15).

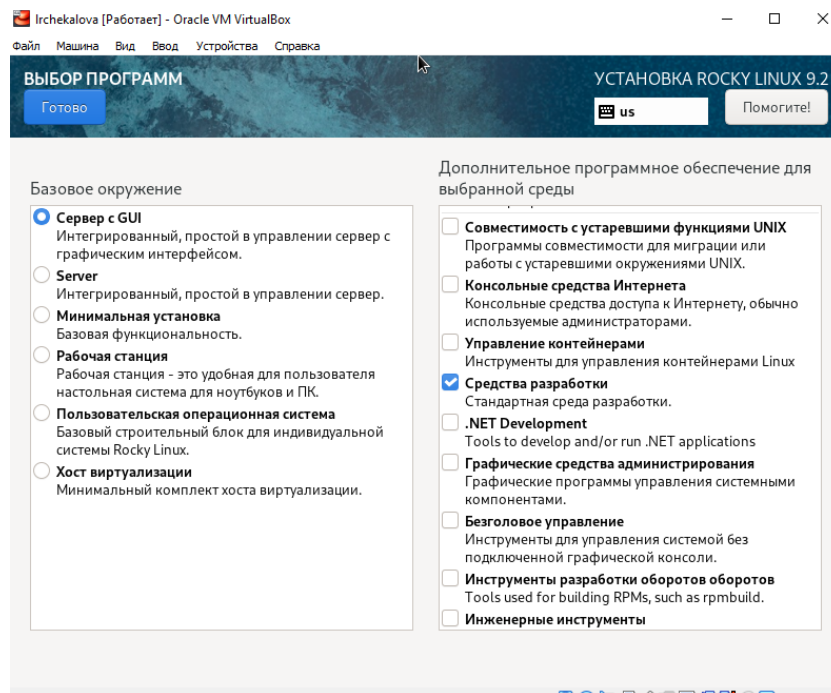


Рис. 0.15: Выбор программ

Включили сетевое соединение и настроили имя узла (рис. 0.16).



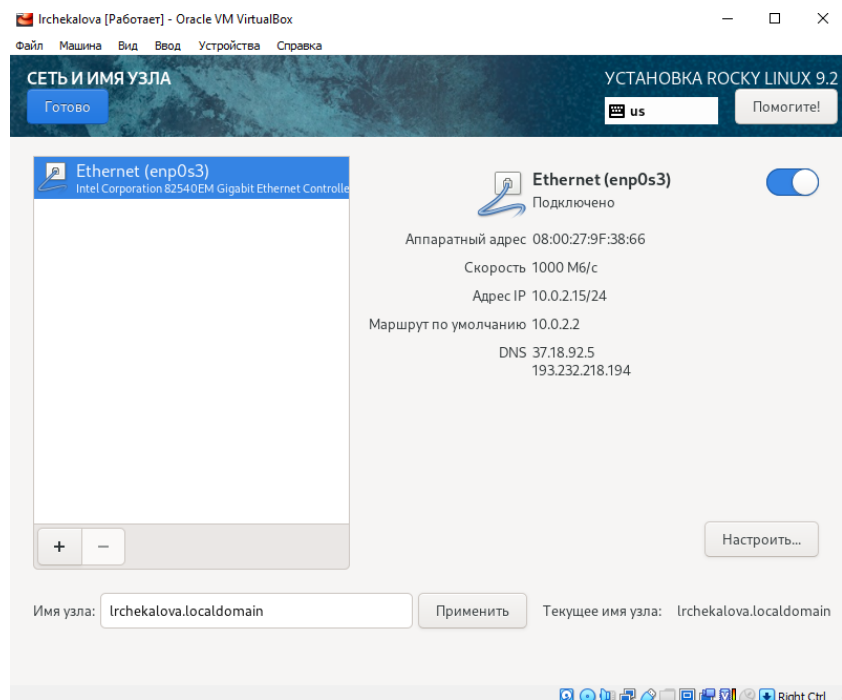


Рис. 0.16: Настройка имени узла

Задали пароль для root (рис. 0.17).

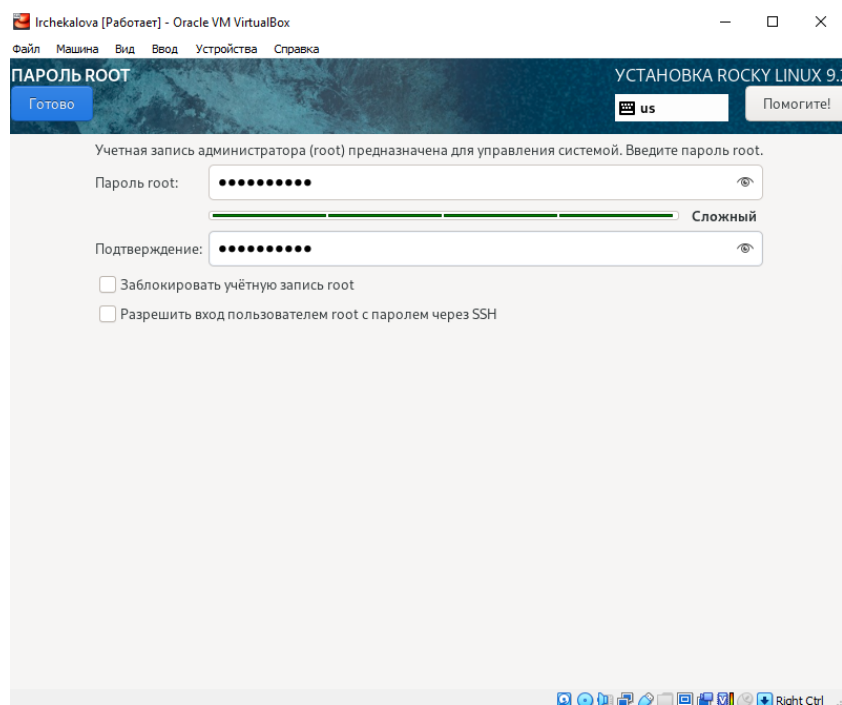


Рис. 0.17: Задание пароля для root

Отключили KDUMP. Окно обзора установки выглядит следующим образом (рис. 0.18).

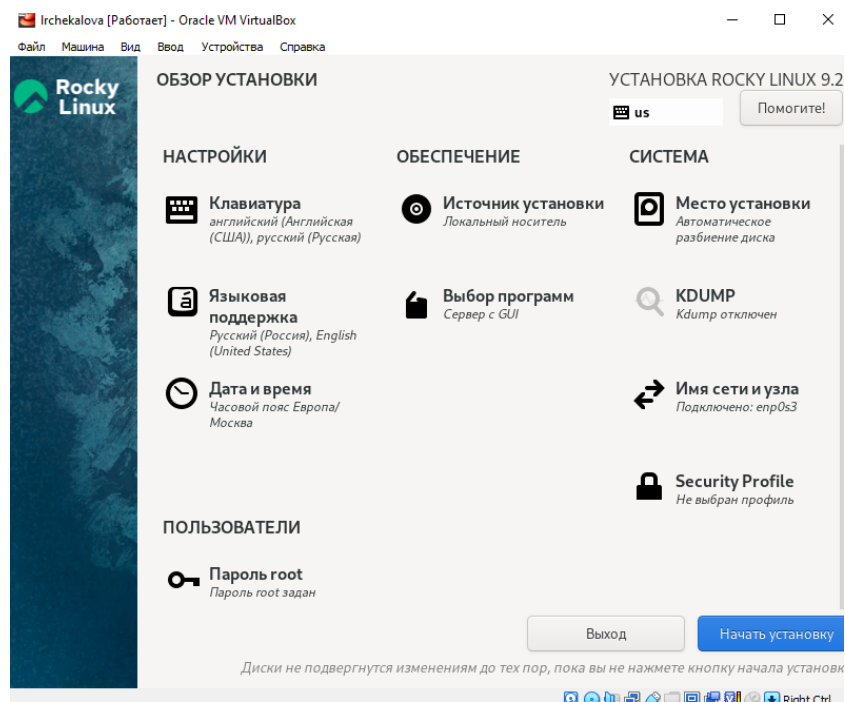


Рис. 0.18: Обзор установки

Перезагрузили систему, залогинились и подключили образ диска дополнений гостевой ОС (рис. 0.19).

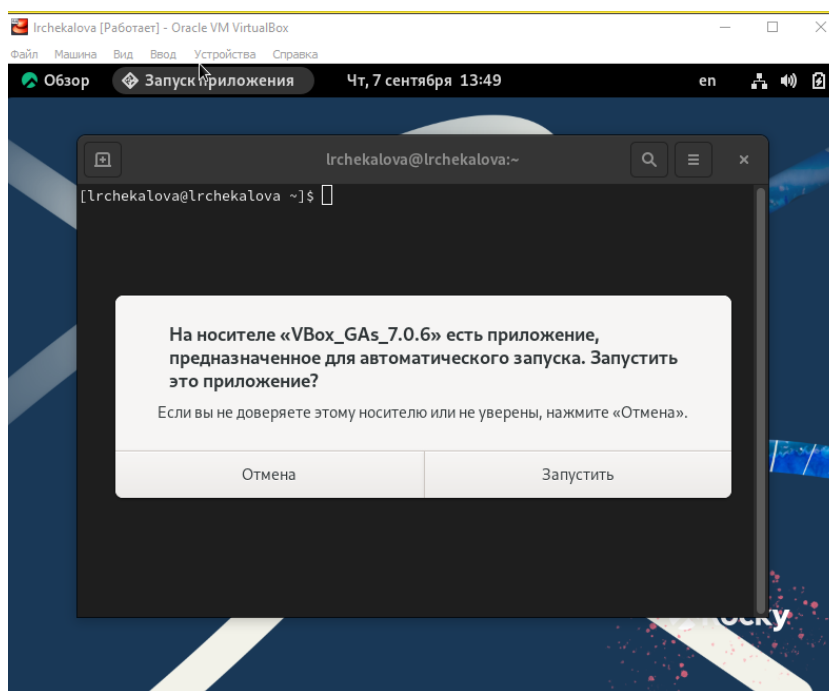


Рис. 0.19: Подключение образа диска дополнений гостевой ОС

С помощью команды `dmesg` нашли информацию о системе, а именно о версии ядра Linux, частоте процессора, модели процессора, объеме доступной оперативной памяти, типе обнаруженного гипервизора (рис. 0.20), типе файловой системы и последовательно-сти монтирования файловых систем (рис. 0.21).

```
lrchekalova@lrchekalova:~$ dmesg | grep "Linux version"
[ 0.000000] Linux version 5.14.0-284.11.1.el9_2.x86_64 (mockbuild@iad1-prod-b
uild001.bld.equ.rockylinux.org) (gcc (GCC) 11.3.1 20221121 (Red Hat 11.3.1-4), G
NU ld version 2.35.2-37.el9) #1 SMP PREEMPT_DYNAMIC Tue May 9 17:09:15 UTC 2023
lrchekalova@lrchekalova:~$ dmesg | grep "Detected Mhz processor"
lrchekalova@lrchekalova:~$ dmesg | grep "processor"
bash: processor: command not found...
lrchekalova@lrchekalova:~$ dmesg | grep "processor"
[ 0.000018] tsc: Detected 2294.800 MHz processor
[ 0.309099] smpboot: Total of 1 processors activated (4589.60 BogoMIPS)
lrchekalova@lrchekalova:~$ dmesg | grep "CPU0"
[ 0.307917] smpboot: CPU0: Intel(R) Core(TM) i7-3610QM CPU @ 2.30GHz (family:
0x6, model: 0x3a, stepping: 0x9)
lrchekalova@lrchekalova:~$ dmesg | grep "Memory available"
lrchekalova@lrchekalova:~$ dmesg | grep "Memory"
[ 0.042191] Memory: 260860K/2096696K available (14342K kernel code, 5536K rwd
ata, 10180K rodata, 2792K init, 7524K bss, 143480K reserved, 0K cma-reserved)
[ 0.347932] x86/mm: Memory block size: 128MB
lrchekalova@lrchekalova:~$ dmesg | grep "Hypervisor"
[ 0.000000] Hypervisor detected: KVM
```

Рис. 0.20: Поиск информации с помощью `dmesg`, 1

```
[lrchekalova@lrchekalova ~]$ dmesg | grep "Filesystem"
[   8.864816] XFS (dm-0): Mounting V5 Filesystem
[  17.929236] XFS (sda1): Mounting V5 Filesystem
[lrchekalova@lrchekalova ~]$
```

Рис. 0.21: Поиск информации с помощью dmesg, 2

## **Выводы**

В результате лабораторной работы мной были получены знания об установке операционной системы на виртуальную машину, отточены навыки работы с системой Git, а также согласно установленному шаблону был создан рабочий каталог для лабораторных работ.

## Список литературы

1. Установка и конфигурация ОС на виртуальную машину [Электронный ресурс]. URL: [https://esystem.rudn.ru/pluginfile.php/2090270/mod\\_folder/content/0/001-lab\\_virtualbox.pdf?forcedownload=1](https://esystem.rudn.ru/pluginfile.php/2090270/mod_folder/content/0/001-lab_virtualbox.pdf?forcedownload=1).
2. Управление версиями [Электронный ресурс]. URL: [https://esystem.rudn.ru/pluginfile.php/2090270/mod\\_folder/content/0/002-lab\\_vcs.pdf?forcedownload=1](https://esystem.rudn.ru/pluginfile.php/2090270/mod_folder/content/0/002-lab_vcs.pdf?forcedownload=1).