

Лабораторная работа №1 и №2

Установка и конфигурация операционной системы на виртуальную
машину и управление версиями

Кекишева Анастасия Дмитриевна

Содержание

1	Цель работы	6
2	Задание	7
3	Теоретическое введение	8
4	Выполнение лабораторной работы	10
5	Выводы	30
6	Список литературы	31

Список таблиц

Список иллюстраций

4.1	Имя и тип ОС	10
4.2	Объём памяти	11
4.3	Создание нового жёсткого диска	12
4.4	Тип диска - VDI	12
4.5	Формат хранения - динамический	13
4.6	Определение размера диска	13
4.7	Добавление оптического диска и выбор образа CentOS	14
4.8	Настройка клавиатуры	15
4.9	Отключение KDUMP	16
4.10	Настройка базового окружения	17
4.11	Настройка сетевого узла	18
4.12	Пароль для root	19
4.13	Создание пользователя с правами администратора	20
4.14	Принятие лицензионного соглашения	21
4.15	Установка драйверов	22
4.16	Выполнение команды gmesg	23
4.17	Нахождение версии ядра Linux, частоты и модели	23
4.18	Нахождение оперативной памяти и тип гипервизора	24
4.19	Иерархия файловых систем	24
4.20	Выполнение команды mount	24
4.21	Установка git-flow	25
4.22	Скачивание пакетов gh	25

4.23	Установка пакета gh	25
4.24	Базовая настройка git и команда создания клча SSH	26
4.25	Ключь SSH	26
4.26	Команда генерации RSA ключа	27
4.27	RSA ключ	27
4.28	Настройка автоматических подписей коммитов	27
4.29	Автораизация на github	28
4.30	Создание репозитория курса на основе шаблона	28
4.31	Удаление файлов json и создание папок для лабораторных	28
4.32	Отправка файлов в репозиторий	29

1 Цель работы

- Приобретение практических навыков установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.
- Изучение идеологии и применение средств контроля версий.
- Освоение работы с git.

2 Задание

Для лабораторной №1:

Получить следующую информацию: - Версия ядра Linux (Linux version). - Частота процессора (Detected Mhz processor). - Модель процессора (CPU0). - Объем доступной оперативной памяти (Memory available). - Тип обнаруженного гипервизора (Hypervisor detected). - Тип файловой системы корневого раздела.

Для лабораторной №2:

- 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.

3 Теоретическое введение

Опишем теоретические аспекты, связанные с выполнением работы.

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.

С помощью встроенной утилиты «gh» можно создавать и разбирать пул-реквесты, включая работу в команде, заводить и просматривать сообщения об ошибках (issue), клонировать репозитории, просматривать различия в коде, выполнять рецензирование изменений и управлять формированием релизов на GitHub.

4 Выполнение лабораторной работы

1. Скачала виртуальную машину и дистрибутив Centos
2. Создала новую виртуальную машину, указала имя виртуальной машины – adkekisheva. Выбрала тип операционной системы — Linux, RedHat (рис. @fig:001).

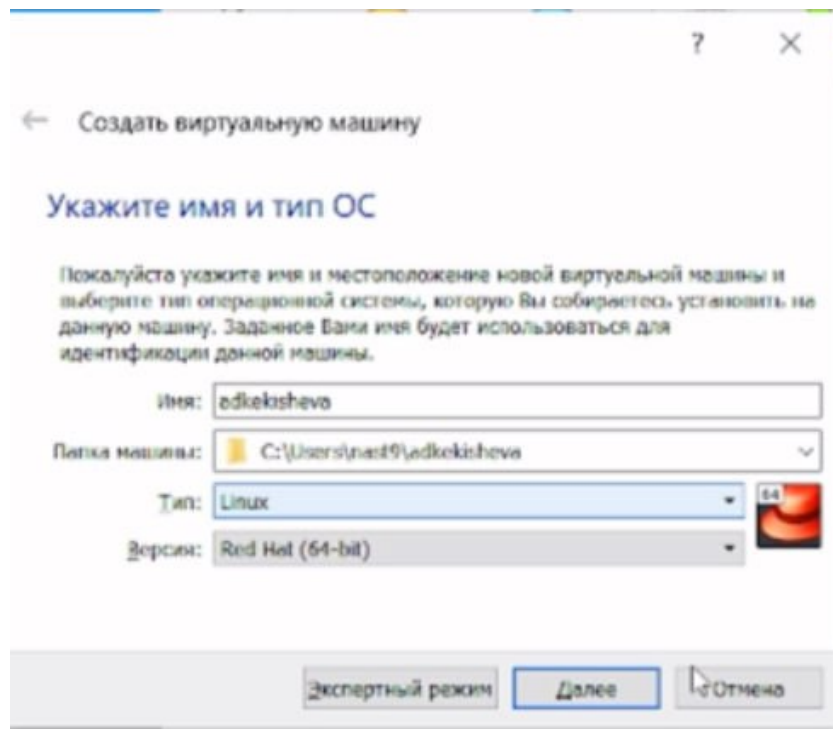


Рис. 4.1: Имя и тип ОС

3. Указала размер основной памяти виртуальной машины — 2048 МБ (рис. @fig:002).

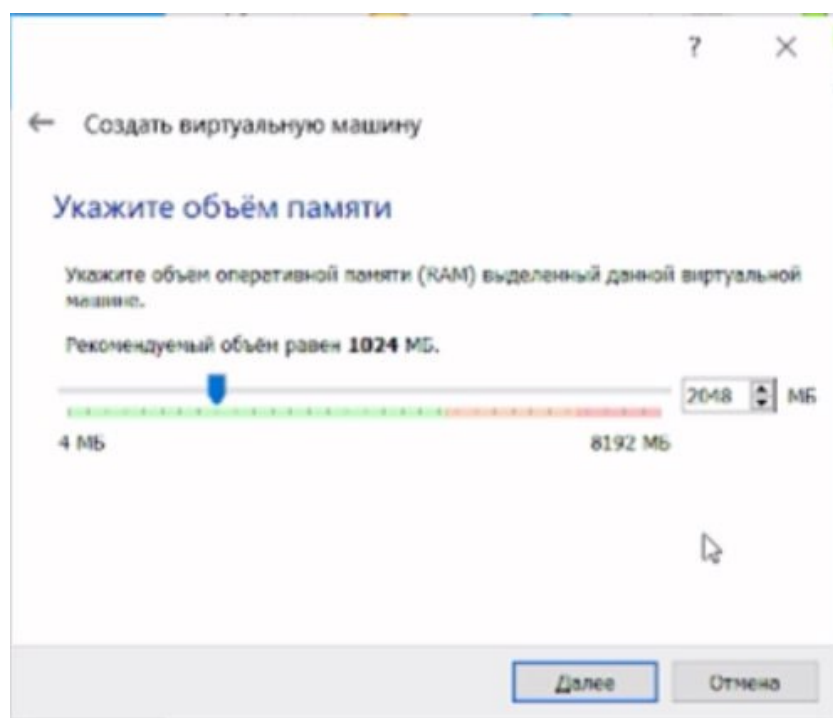


Рис. 4.2: Объём памяти

4. Создала новый виртуальный диск (рис. @fig:003), задавала конфигурацию жёсткого диска — загрузочный VDI (рис. @fig:004), выбрала динамический виртуальный диск (рис. @fig:005).

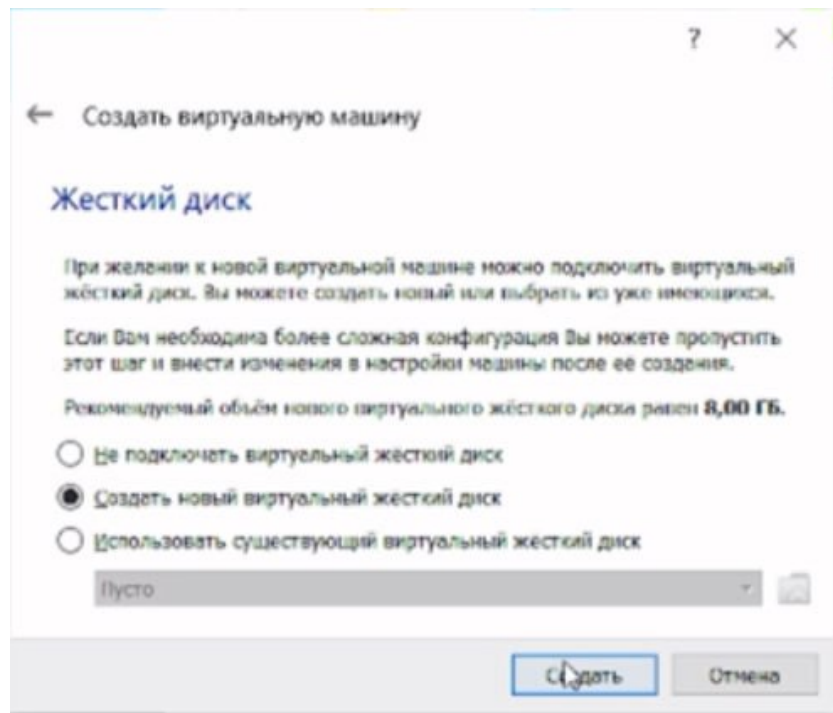


Рис. 4.3: Создание нового жёского диска

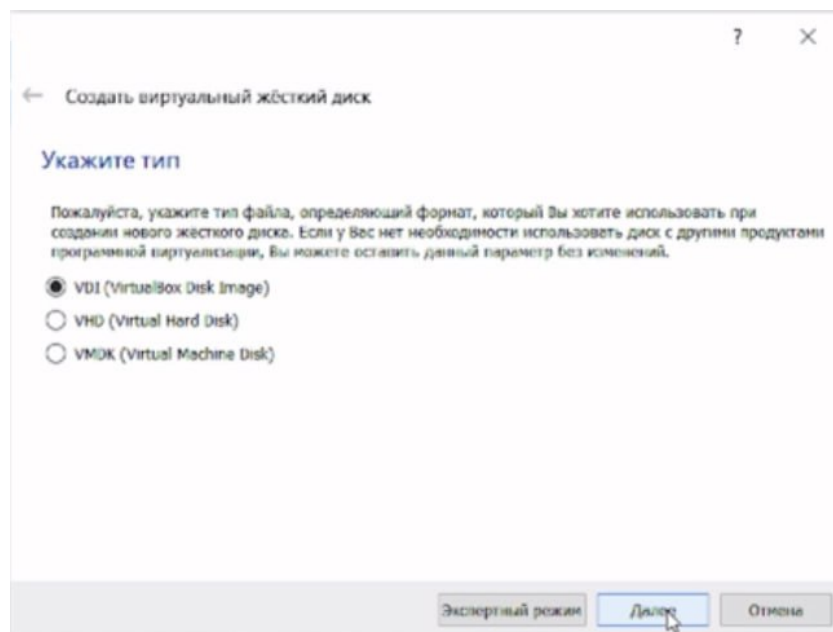


Рис. 4.4: Тип диска - VDI

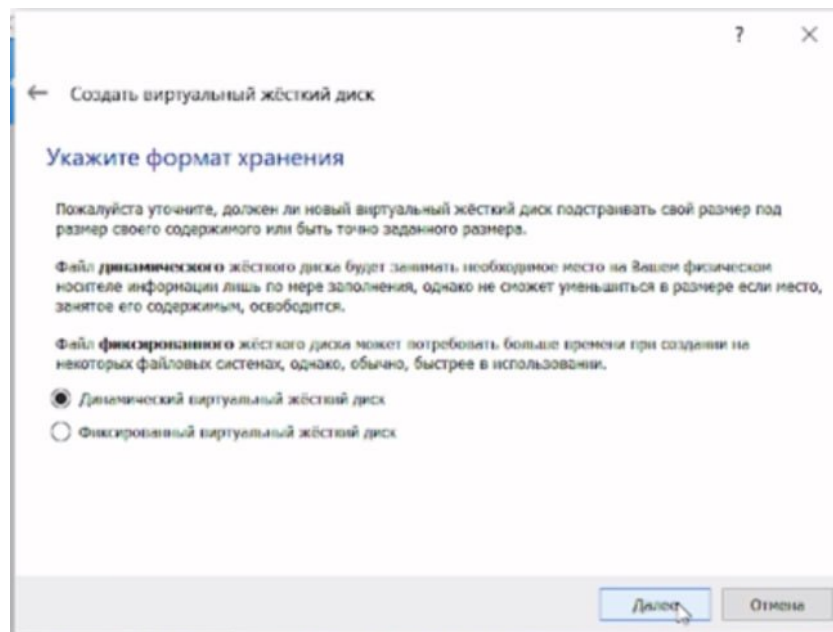


Рис. 4.5: Формат хранения - динамический

5. Задала размер диска — 40 ГБ и его расположение (рис. @fig:006).

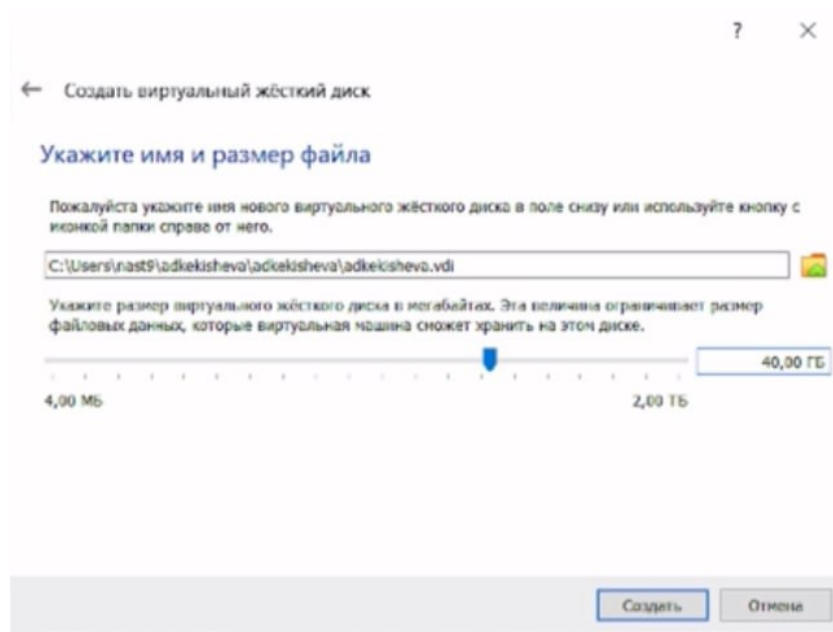


Рис. 4.6: Определение размера диска

6. Добавила новый привод оптических дисков и выбрала образ операционной системы (рис. @fig:007).

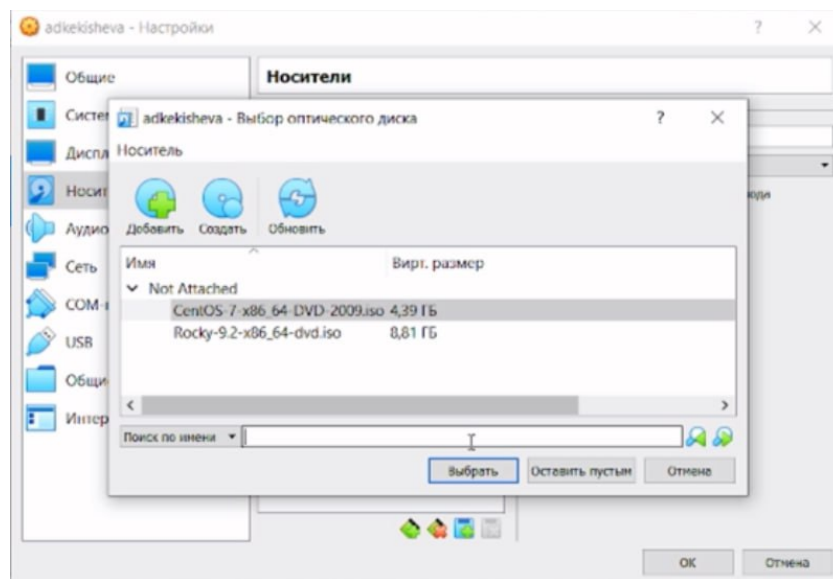


Рис. 4.7: Добавление оптического диска и выбор образа CentOS

7. Запустила виртуальную машину, скорректировала часовой пояс, раскладку клавиатуры (рис. @fig:008).

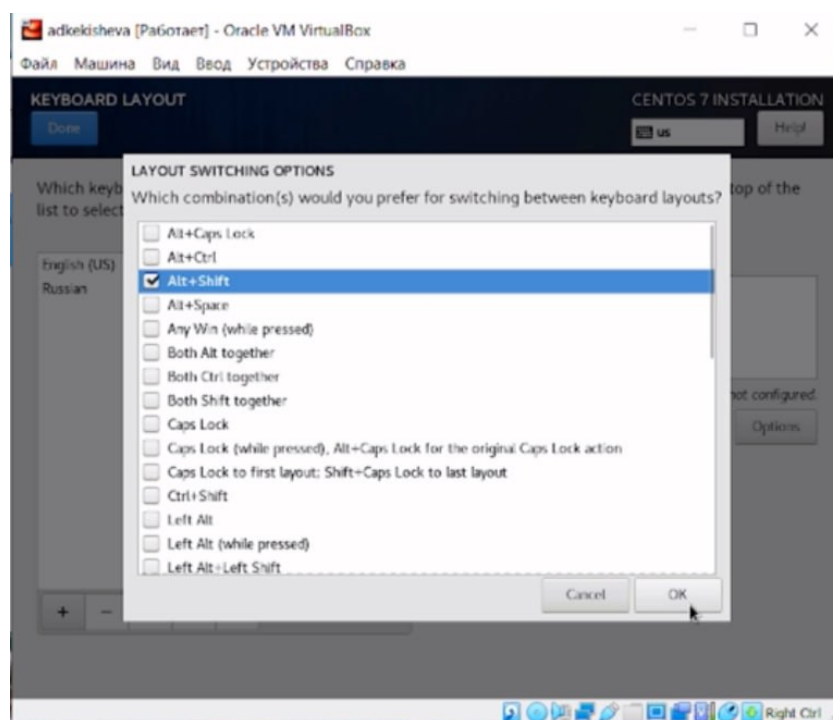


Рис. 4.8: Настройка клавиатуры

8. Отключила KDUMP (рис. @fig:009).

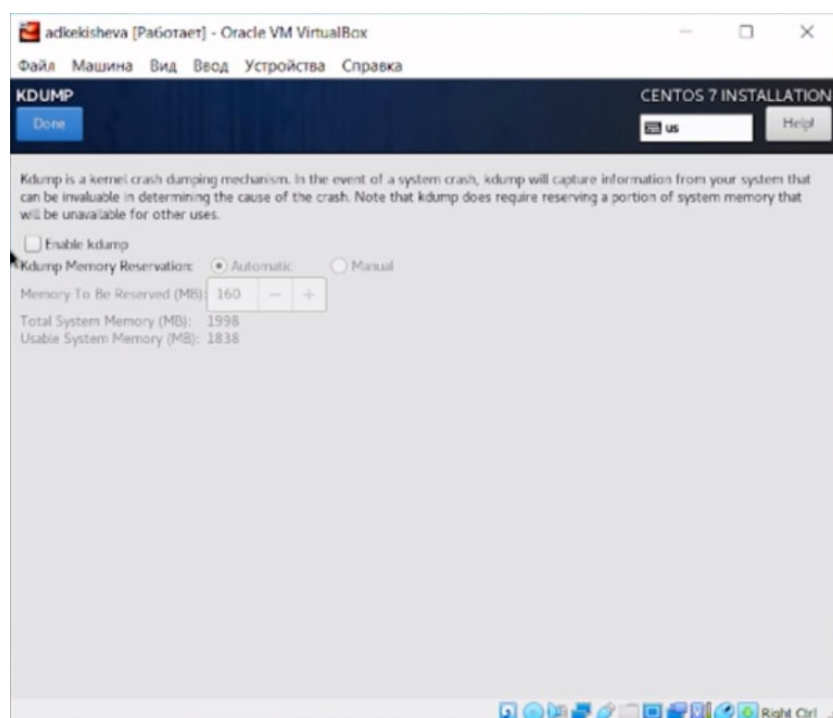


Рис. 4.9: Отключение KDUMP

9. Указала в качестве базового окружения Server with GUI , а в качестве дополнения — Development Tools (рис. @fig:010).

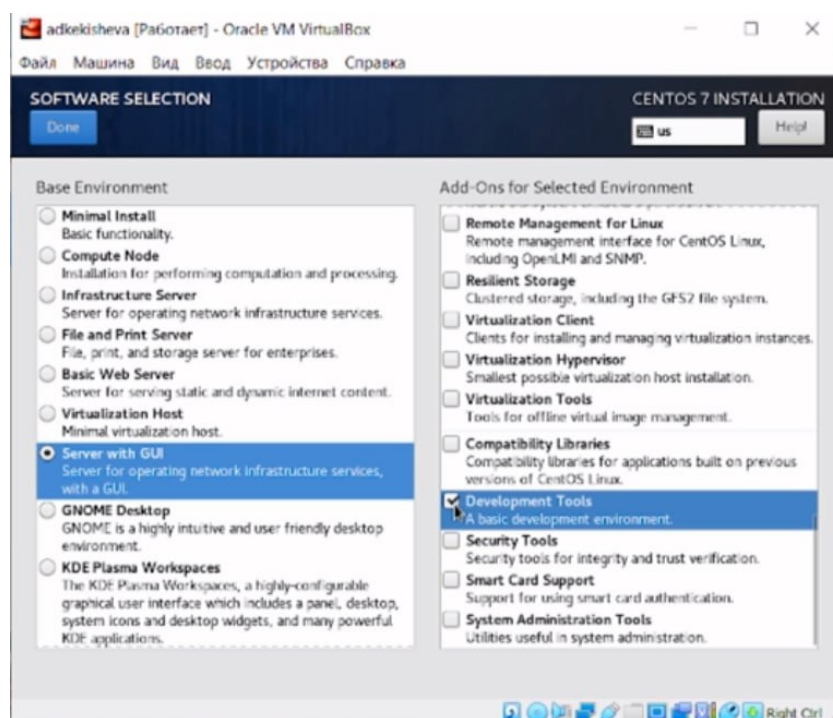


Рис. 4.10: Настройка базового окружения

10. Включила сетевое соединение и в качестве имени узла указала `adkekisheva.localdomain` (рис. @fig:011).

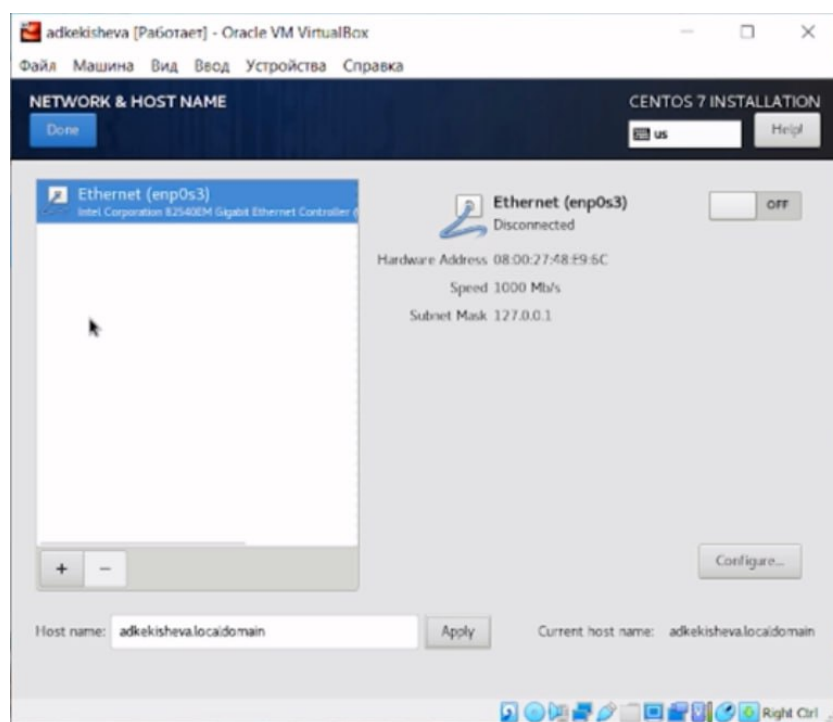


Рис. 4.11: Настройка сетевого узла

11. Установила пароль для root (рис. @fig:012) и пользователя с правами администратора (рис. @fig:013).

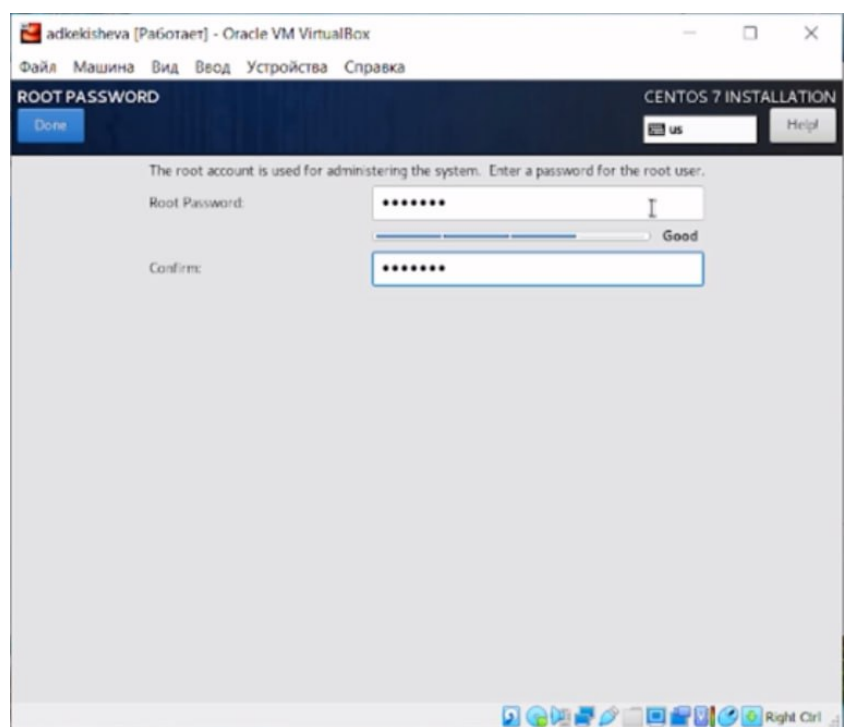


Рис. 4.12: Пароль для root

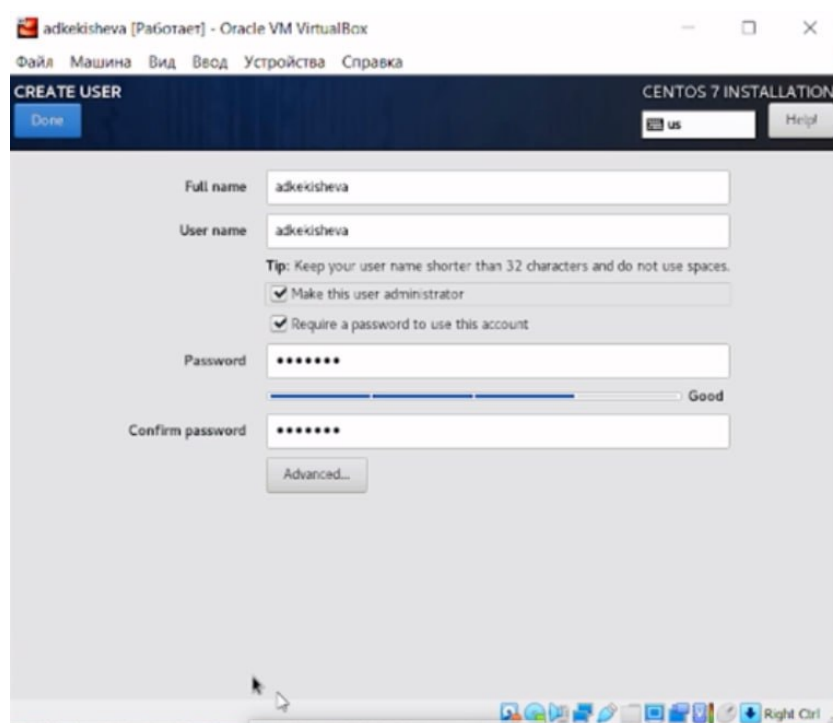


Рис. 4.13: Создание пользователя с правами администратора

12. После завершения установки операционной системы перезапустила виртуальную машину и приняла условия лицензии (рис. @fig:011).

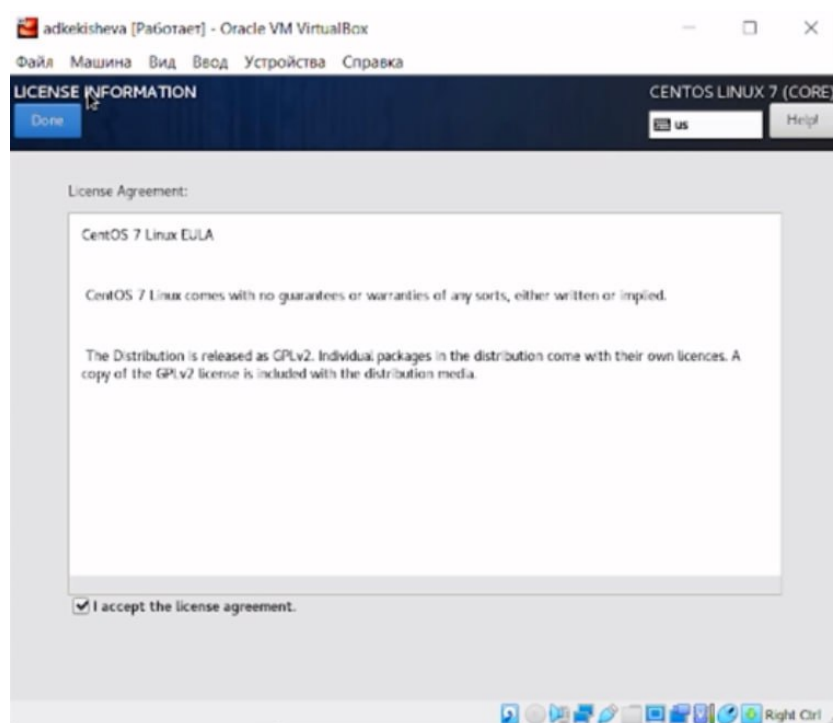


Рис. 4.14: Принятие лицензионного соглашения

13. Вошла в ОС под заданной учётной записью и установила драйверы (рис. @fig:015).

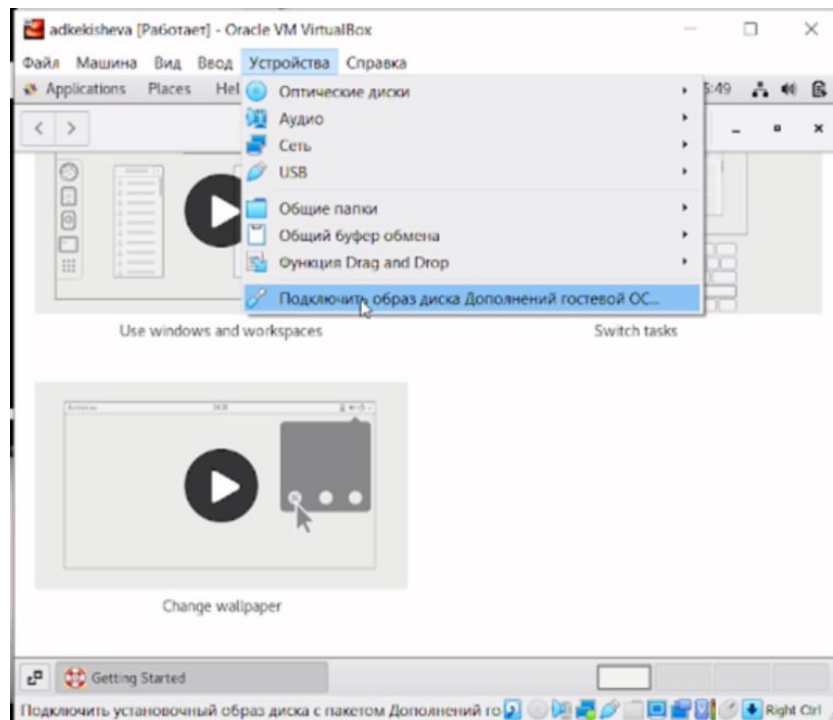


Рис. 4.15: Установка драйверов

Далее перешла к анализу последовательности загрузки системы.

14. Выполнила команду `dmesg` в консоли (рис. @fig:016) и получилось, что загрузка происходит в следующей последовательности: 1 - загрузка и инициализация ядра; 2 - обнаружение и конфигурирование устройств; 3 - создание процессов ядра; 4 - выполнение сценариев запуска; 5 - работа в многопользовательском режиме.

```
adkekisheva@adkekisheva:~$ dmesg
[ 0.000000] Initializing cgroup subsys cpuset
[ 0.000000] Initializing cgroup subsys cpu
[ 0.000000] Initializing cgroup subsys cpuct
[ 0.000000] Linux version 3.10.0-1160.el7.x86_64 (mockbuild@kbuilder.bsys.centos.org) (gcc version 4.8.5 20150623 (Red Hat 4.8.5-44) (GCC) ) #1 SMP Mon Oct 19 16:18:59 UTC 2020
[ 0.000000] Command line: BOOT_IMAGE=/vmlinuz-3.10.0-1160.el7.x86_64 root=/dev/mapper/centos_adkekisheva-root ro spectre_v2=retpoline rd.lvm.lv=centos_adkekisheva/root rd.lvm.lv=centos_adkekisheva/swap r hgb quiet LANG=en_US.UTF-8
[ 0.000000] e820: BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x0000000000000bfff] usable
[ 0.000000] BIOS-e820: [mem 0x0000000000000bfc00-0x0000000000000ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000000f0000-0x0000000000000ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000100000-0x00000000007ffff] usable
[ 0.000000] BIOS-e820: [mem 0x00000000007ff000-0x00000000007ffff] ACPI data
[ 0.000000] BIOS-e820: [mem 0x000000000fec00000-0x000000000fec0ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x000000000fee00000-0x000000000fee0ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x000000000ffffc0000-0x000000000ffffffffff] reserved
[ 0.000000] NX (Execute Disable) protection: active
[ 0.000000] SMBIOS 2.5 present.
[ 0.000000] DMI: Innatek GmbH VirtualBox/VirtualBox, BIOS VirtualBox 12/01/2006
[ 0.000000] Hypervisor detected: KVM
[ 0.000000] e820: update [mem 0x00000000-0x00000ffff] usable ==> reserved
[ 0.000000] e820: remove [mem 0x00000000-0x0000ffff] usable
[ 0.000000] e820: last pfn = 0x7ffff max arch pfn = 0x400000000
[ 0.000000] MTRR default type: uncachable
[ 0.000000] MTRR variable ranges disabled:
[ 0.000000] PAT configuration [0-7]: WB WC UC- UC WB WP UC- UC
[ 0.000000] CPU MTRRs all blank - virtualized system
[ 0.000000] found SMP MP-table at [mem 0x0009ffff-0x0009ffff] mapped at [ffffffffff200ff0]
[ 0.000000] Base memory trampoline at [ffff6c76c0099000] 99000 size 24576
[ 0.000000] BRK [0x5c274000, 0x5c274fff] PGTABLE
[ 0.000000] BRK [0x5c275000, 0x5c275fff] PGTABLE
[ 0.000000] BRK [0x5c276000, 0x5c276fff] PGTABLE
[ 0.000000] BRK [0x5c277000, 0x5c277fff] PGTABLE
```

Рис. 4.16: Выполнение команды dmesg

15. С помощью команды `dmesg | grep -i version` получила версию ядра Linux, которая равна 3.10.0 (рис. @fig:017).
16. С помощью команды `dmesg | grep -i Mhz` получила частоту процессора, равную 2112.004 Mhz (рис. @fig:017).
17. С помощью команды `dmesg | grep -i CPU` получила модель процессора (CPU0) (рис. @fig:017).

```
adkekisheva@adkekisheva:~$ dmesg | grep -i version
[ 0.000000] Linux version 3.10.0-1160.el7.x86_64 (mockbuild@kbuilder.bsys.centos.org) (gcc version 4.8.5 20150623 (Red Hat 4.8.5-44) (GCC) ) #1 SMP Mon Oct 19 16:18:59 UTC 2020
[ 0.000000] IOAPIC[0]: apic id 1, version 32, address 0xfec00000, GSI 0-23
[ 0.280535] acpihp: ACPI Hot Plug PCI Controller Driver version: 0.5
[ 0.880296] Block layer SCSI generic (bsg) driver version 0.4 loaded (major 248)
[ 0.880444] pci_hotplug: PCI Hot Plug PCI Core version: 0.5
[ 0.880448] pciehp: PCI Express Hot Plug Controller Driver version: 0.4
[ 0.880455] shpchp: Standard Hot Plug PCI Controller Driver version: 0.4
[ 0.882213] crash memory driver: version 1.1
[ 0.939050] registered taskstats version 1
[ 1.116702] fuse init (API version 7.23)
[ 1.317826] device-mapper: uevent: version 1.0.3
[ 1.525556] e1000: Intel(R) PRO/1000 Network Driver - version 7.3.21-k8-NAPI
[ 1.639608] libata version 3.00 loaded.
[ 2.125200] ata_piix 0000:00:01:1: version 2.13
[ 2.129692] ahci 0000:00:0d:0: version 3.0
[ 4.407953] vboxguest: Successfully loaded version 6.1.46 r158378
[ 4.407132] vboxguest: Successfully loaded version 6.1.46 r158378 (interface 0x00010004)
[ 4.591340] AVX2 version of gcm enc/dec engaged.
[ 12.226468] nf_conntrack version 0.5.0 (16384 buckets, 65536 max)
[ 21.474672] 12:54:23.031427 main 05 Version: #1 SMP Mon Oct 19 16:18:59 UTC 2020
[ 28345.109083] nf_conntrack version 0.5.0 (16384 buckets, 65536 max)
adkekisheva@adkekisheva:~$ dmesg | grep -i Mhz
[ 0.000000] tsc: Detected 2112.004 MHz processor
[ 1.860703] tsc: Refined TSC clocksource calibration: 2100.000 MHz
[ 2.125107] e1000 0000:00:0d:0:0: eth0: (PCI:33MHz:32-bit) 88:00:27:48:e9:6c
adkekisheva@adkekisheva:~$ dmesg | grep -i CPU0
[ 0.178086] smpboot: CPU0: Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz (fam: 06, model: 8e, stepping: 0c)
```

Рис. 4.17: Нахождение версии ядра Linux, частоты и модели

18. С помощью команды `dmesg | grep -i memory` узнала, что объем доступной оперативной памяти составляет 2013192k (рис. @fig:018).

19. С помощью команды `dmesg | grep -i hypervisor` нашла тип обнаруженного гипервизора, это – KVM (рис. @fig:018).

```
adkekisheva@adkekisheva:~  
File Edit View Search Terminal Help  
[adkekisheva@adkekisheva ~]$ dmesg | grep -i memory  
[ 0.000000] Base memory trampoline at [ffff8c76c0090000] 99000 size 24576  
[ 0.000000] Early memory node ranges  
[ 0.000000] PM: Registered nosave memory: [mem 0x0009f000-0x0009ffff]  
[ 0.000000] PM: Registered nosave memory: [mem 0x000a0000-0x000affff]  
[ 0.000000] PM: Registered nosave memory: [mem 0x000f0000-0x000fffff]  
[ 0.000000] Memory: 2013192K/2097088K available (7708K kernel code, 392K absent, 83504K reserved, 5954K data, 1084K init)  
[ 0.000000] please try 'cgroup.disable_memory' option if you don't want memory cgroups  
[ 0.062500] Initializing cgroup subsys memory  
[ 0.277140] x86/mm: Memory block size: 128MB  
[ 0.846040] Freeing initrd memory: 30816K freed  
[ 0.882019] Non-volatile memory driver v1.3  
[ 0.882213] crash memory driver: version 1.1  
[ 0.944000] Freeing unused kernel memory: 1004K freed  
[ 0.945109] Freeing unused kernel memory: 392K freed  
[ 0.946159] Freeing unused kernel memory: 536K freed  
[ 2.132765] [drm] Max dedicated hypervisor surface memory is 507904 KiB  
[ 2.132766] [drm] Maximum display memory size is 16384 KiB  
[ 2.133765] [TTM] Zone kernel: Available graphics memory: 1023474 KiB  
[adkekisheva@adkekisheva ~]$ dmesg | grep -i hypervisor  
[ 0.000000] Hypervisor detected: KVM  
[ 2.132765] [drm] Max dedicated hypervisor surface memory is 507904 KiB
```

Рис. 4.18: Нахождение оперативной памяти и тип гипервизора

20. С помощью команды `lsblk -f` узнала тип файловой системы корневого раздела – xfs (рис. @fig:019).

```
[adkekisheva@adkekisheva ~]$ lsblk -f  
NAME        FSTYPE LABEL        UUID                               MOUNTPOINT  
sda  
├─sda1 xfs          d5cd7575-8ffa-4795-9db4-14416bd938f8 /boot  
├─sda2 LVM2 mem      T6v7hT-4p4C-880K-daHf-mPIJ-bzrw-b2LNVN  
├─centos_adkekisheva-root  
│   xfs          2d4557c2-0e00-47bf-8a94-69e50e7e7b13 /  
└─centos_adkekisheva-swap  
    swap        954e3273-a52f-4575-b62e-6d56d22cc254 [SWAP]  
sr0  
sr1 iso9660    VBox_GAs_6.1.46 2023-07-12-17-05-32-49 /run/media
```

Рис. 4.19: Иерархия файловых систем

21. Последовательность монтирования файловых систем посмотрела командой `mount` (рис. @fig:020).

```
[adkekisheva@adkekisheva ~]$ mount  
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)  
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)  
devtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=1006608K,nr_inodes=251652,mode=755)  
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)  
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,seclabel)  
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,seclabel,gid=5,mode=620,ptmxmode=000)  
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,mode=755)  
tmpfs on /sys/fs/cgroup type tmpfs (ro,nosuid,nodev,noexec,seclabel,mode=755)  
cgroup on /sys/fs/cgroup/systemd type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,xattr,release_agent=/usr/lib/systemd/systemd-cgroups-agent,hostname=systemd)  
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime)  
cgroup on /sys/fs/cgroup/blkio type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,blkio)  
cgroup on /sys/fs/cgroup/net_cls,net_prio type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,net_prio,net_cls)  
cgroup on /sys/fs/cgroup/cpu,cpuacct type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,cpuacct,cpu)  
cgroup on /sys/fs/cgroup/perf_event type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,perf_event)  
cgroup on /sys/fs/cgroup/hugetlb type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,hugetlb)  
cgroup on /sys/fs/cgroup/devices type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,devices)  
cgroup on /sys/fs/cgroup/freezer type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,freezer)  
cgroup on /sys/fs/cgroup/cpuset type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,cpuset)  
cgroup on /sys/fs/cgroup/pids type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,pids)  
cgroup on /sys/fs/cgroup/memory type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,memory)
```

Рис. 4.20: Выполнение команды mount

На этом часть лабораторной выполнена, переходим ко второй части.

1. Для начала установила программное обеспечение git-flow (рис. @fig:021).

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

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

#1) Respect the privacy of others.
#2) Think before you type.
#3) With great power comes great responsibility.

[sudo] password for adkeki:
Sorry, try again.
[sudo] password for adkeki:
## git-flow no-make installer ##
Installing git-flow to /usr/local/bin
Cloning repo from github to gitflow
Cloning into 'gitflow'...
remote: Enumerating objects: 4270, done.
remote: Total 4270 (delta 0), reused 0 (delta 0), pack-reused 4270
Receiving objects: 100% (4270/4270), 1.74 MiB | 2.37 MiB/s, done.
Resolving deltas: 100% (2533/2533), done.
```

Рис. 4.21: Установка git-flow

2. Установила gh, однако обычной командой он у меня не скачивался, поэтому скачала, через пакет (рис. @fig:022 и @fig:023).

```
[adkeki@adkeki$ tmp]$ wget https://github.com/cli/cli/releases/download/v2.13.0/gh_2.13.0_linux_386.rpm
--2023-09-07 10:14:09-- https://github.com/cli/cli/releases/download/v2.13.0/gh_2.13.0_linux_386.rpm
Resolving github.com (github.com)... 140.82.121.4
Connecting to github.com (github.com)|140.82.121.4|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/212613049/c380f973-5455-433d-87cb-995c057c240
8?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAK4CV5H53ALZF2023090712FUS-EAST-1%2F%3A2Faws4_request&X-Amz-Date=20230
907140820&X-Amz-Expires=300&X-Amz-Signature=6c31c0ddbb3ba0e2b05008e2c847a3167c90991bbaad9b0f42b4e990b7d&X-Amz-SignedHeaders=sh
ostfactor;idmkey;idmrepo;idmrepo_response-content-disposition;attachment;38020f11eame30gh_2.13.0_linux_386.rpm&response-c
ontent-type=application%2Foctet-stream [following]
--2023-09-07 10:14:09-- https://objects.githubusercontent.com/github-production-release-asset-2e65be/212613049/c380f973-5455-433d-8
7cb-995c057c2408?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAK4CV5H53ALZF2023090712FUS-EAST-1%2F%3A2Faws4_request&X
-Amz-Date=20230907140820&X-Amz-Expires=300&X-Amz-Signature=6c31c0ddbb3ba0e2b05008e2c847a3167c90991bbaad9b0f42b4e990b7d&X-Amz-
SignedHeaders=shostfactor;idmkey;idmrepo;idmrepo_response-content-disposition;attachment;38020f11eame30gh_2.13.0_linux_386
.rpm&response-content-type=application%2Foctet-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.111.133, 185.199.108.133, 185.199.109.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.111.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 7287441 (6.9M) [application/octet-stream]
Saving to: 'gh_2.13.0_linux_386.rpm'

100%[=====] 7,287,441 6.35MB/s In 1.1s

2023-09-07 10:14:11 (6.35 MB/s) - 'gh_2.13.0_linux_386.rpm' saved [7287441/7287441]
```

Рис. 4.22: Скачивание пакетв gh

```
[adkeki@adkeki$ tmp]$ sudo yum localinstall gh_2.13.0_linux_386.rpm
Loaded plugins: fastestmirror, langpacks
Examining gh_2.13.0_linux_386.rpm: gh-2.13.0-1.i386
Marking gh_2.13.0_linux_386.rpm to be installed
Resolving Dependencies
--> Running transaction check
--> Package gh.i386 0:2.13.0-1 will be installed
--> Finished Dependency Resolution

base77/x86_64 | 3.6 kB | 00:00
https://crn.r-project.org/package3Dgh/repodata/repond.xml: [Errno 14] HTTPS Error 404 - Not Found
Trying other mirror.
To address this issue please refer to the below wiki article

https://wiki.centos.org/yum-errors

If above article doesn't help to resolve this issue please use https://bugs.centos.org/.

extras77/x86_64 | 2.9 kB | 00:00
updates77/x86_64 | 2.9 kB | 00:00

Dependencies Resolved

=====
Package Arch Version Repository Size
=====
Installing:
gh i386 2.13.0-1 /gh_2.13.0_linux_386 26 M
Transaction Summary
=====
Install 1 Package
```

Рис. 4.23: Установка пакета gh

3. Задала имя и email для моего репозитория, настроила utf-8 в выводе сообщений git, задала имя master для начальной ветки, параметры autocrlf и safecrlf, что

означает, что файлы проверяются как есть, в при коммите конфертируются из CRLF в LF для записи текстовых файлов. Также создала ключ SSH по алгоритму RSA с размером 4096 бит (рис. @fig:024).

```
[adkekisheva@adkekisheva tmp]$ git config --global user.name "adkekisheva"
[adkekisheva@adkekisheva tmp]$ git config --global user.email "1032201194@pfur.ru"
[adkekisheva@adkekisheva tmp]$ git config --global core.quotepath false
[adkekisheva@adkekisheva tmp]$ git config --global init.defaultBranch master
[adkekisheva@adkekisheva tmp]$ git config --global core.autocrlf input
[adkekisheva@adkekisheva tmp]$ git config --global core.safecrlf warn
[adkekisheva@adkekisheva tmp]$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/adkekisheva/.ssh/id_rsa): key
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in key.
Your public key has been saved in key.pub.
The key fingerprint is:
SHA256:5xJpG8IP/8qcfq9ebK9EVwQlnVpm1x61xmCdklfox0 adkekisheva@adkekisheva.localdomain
The key's randomart image is:
+--[RSA 4096]-----+
|  +. . . . .B=.oE |
| = 0 0 .04 =+ = |
| + = . 0 . +.B+ |
| = . . . =.+ |
| = 5 . . . |
| 0 0 + . . |
| = + . |
| . 0 |
| .ooo. |
+---[SHA256]-----+
```

Рис. 4.24: Базовая настройка git и команда создания клча SSH

4. Просмотрела созданный ключ SSH (рис. @fig:025).

```
[adkekisheva@adkekisheva tmp]$ cat key
-----BEGIN RSA PRIVATE KEY-----
MIIEKQIBAAKCAgEA7kXanf2kGee57cYi4XqbABH299k3m0bE/gALXAFnIppYNA4v
TSnexjMKMfRczdi/jeSi36MWCCvB7v9zGtMts8X/X0b+A38fCJbPe87g08oDtK67
E4j+QuF1xP32dMDRmvNL47rG0Z3XHJBx4HEdEvSuZTp34kliAgBwBmrpKp76HDv
bo8EkQXEiFc9GJEtLU05mCLDXklkAsUyh8F1Kosq9q8onigXPH4pBzHfJZTU7AAh
eh0kP0xEVeHnpNFH7UMtbvv6p5K/3TwYG7+GwNmUOP6/r/AFvA1CoWwVcDPuJKn
DK6retA2dW8Z9jvfHUII7GcAYLQbH/qW0ztbV09T3phSzaJaEnRzenrZP5w5iDHi
zetgX8e2AuReVhJoRXEwdIMKiqlFL5y0Pu4F5vMV0IBLYtTvNhVQtku43I/JuNT
V2vm8VgXCSBy3FXVvp01yfBccunvbAeLRW4ZLeLIIM197nwxy9wKV5/769iqJlSn
SwBUw47hi0Z1JbGDKbHZT409y+Ip2vWoMYpjCUXZMxR9AmxKVUWkcX+8mAqQHiyI
Z6IN3jZebPI4QZRR2pj/XjgJE0WADp/0jUCArF39uXWq3tK9GvqWfauFcXMP9DCd
I3MpsNwpe/Y73oPk71tV0rBKN7eRGh7Kd1JjZc7ozqcyApEVT5ajt/1CiZsCAwEA
-----
```

Рис. 4.25: Ключь SSH

5. Сгенерировала ключ GPG командой gpg --ful-gen-key (рис. @fig:026).

```
[adkekisheva@adkekisheva tmp]$ gpg --gen-key
gpg (GnuPG) 2.0.22; Copyright (C) 2013 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 RSA (default)
  (2) DSA and Elgamal
  (3) DSA (sign only)
  (4) RSA (sign only)
Your selection? 1
RSA keys may 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
    <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
```

Рис. 4.26: Команда генерации RSA ключа

9. Вывела список ключей и скопировала отпечаток приватного ключа (рис. @fig:027)

```
[adkekisheva@adkekisheva tmp]$ gpg --list-secret-keys --keyid-format LONG
/home/adkekisheva/.gnupg/secring.gpg
-----
sec  4096R/32F65E91D0D91AC0 2023-09-07
uid  4096R/32F65E91D0D91AC0 adkekisheva <1032201194@pfur.ru>
ssb  4096R/3C4788B1B703ADB9 2023-09-07
[adkekisheva@adkekisheva tmp]$ gpg --armor --export <32F65E91D0D91AC0> | xclip -sel clip
```

Рис. 4.27: RSA ключ

10. Настроила автоматические подписи коммитов (рис. @fig:028).

```
[adkekisheva@adkekisheva tmp]$ git config --global user.signingkey 32F65E91D0D91AC0
[adkekisheva@adkekisheva tmp]$ git config --global commit.gpgsign true
[adkekisheva@adkekisheva tmp]$ git config --global gpg.program $(which gpg2)
```

Рис. 4.28: Настройка автоматических подписей коммитов

11. Перешла к настройке gh и сначала авторизировалась, для этого ответила на несколько вопросов и подключилась в гит через браузер (рис. @fig:029).

```
[adkekiheva@adkekiheva tmp]$ gh auth login
? What account do you want to log into? GitHub.com
? You're already logged into github.com. Do you want to re-authenticate? Yes
? What is your preferred protocol for Git operations? SSH
? Upload your SSH public key to your GitHub account? /home/adkekiheva/.ssh/id_ed25519.pub
? Title for your SSH key: centos
? How would you like to authenticate GitHub CLI? Login with a web browser

! First copy your one-time code: 1444-AEC9
Press Enter to open github.com in your browser...
This tool has been deprecated, use 'gio open' instead.
See 'gio help open' for more info.

✓ Authentication complete.
- gh config set -h github.com git_protocol ssh
✓ Configured git protocol
HTTP 422: Validation Failed (https://api.github.com/user/keys)
Key is already in use
```

Рис. 4.29: Авторизация на github

12. Создала репозиторий курса на основе шаблона и вместе с тем локальный каталог для выполнения лабораторных работ (рис. @fig:030).

```
[adkekiheva@adkekiheva tmp]$ mkdir -p ~/work/study/2023-2024/"Информационная безопасность"
[adkekiheva@adkekiheva tmp]$ cd ~/work/study/2023-2024/"Информационная безопасность"
[adkekiheva@adkekiheva Информационная безопасность]$ gh repo create study_2023-2024_infosec --template:yanadharma/course-directory-student-template --public
✓ Created repository adkekiheva/study_2023-2024_infosec on GitHub
[adkekiheva@adkekiheva Информационная безопасность]$ git clone --recursive git@github.com:adkekiheva/study_2023-2024_infosec.git infosec
Cloning into 'infosec'...
remote: Enumerating objects: 27, done.
remote: Counting objects: 100% (27/27), done.
remote: Compressing objects: 100% (26/26), done.
remote: Total 27 (delta 1), reused 13 (delta 0), pack-reused 0
Receiving objects: 100% (27/27), 16.93 KiB | 0 bytes/s, done.
Resolving deltas: 100% (1/1), done.
Submodule 'template/presentation' (https://github.com/yanadharma/academic-presentation-markdown-template.git) registered for path 'template/presentation'
Submodule 'template/report' (https://github.com/yanadharma/academic-laboratory-report-template.git) registered for path 'template/report'
Cloning into '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
Unpacking objects: 100% (82/82), done.
Submodule path 'template/presentation': checked out 'b1be3800ee91f5809264cb755d316174540b753e'
Cloning into 'template/report'...
remote: Enumerating objects: 101, done.
```

Рис. 4.30: Создание репозитория курса на основе шаблона

13. Перешла в каталог курса infosec, удалила в нём лишние файлы и создала необходимые каталоги (рис. @fig:031).

```
[adkekiheva@adkekiheva Информационная безопасность]$ cd infosec
[adkekiheva@adkekiheva infosec]$ rm package.json
[adkekiheva@adkekiheva infosec]$ make COURSE=infosec
[adkekiheva@adkekiheva infosec]$ git add .
warning: You ran 'git add' with neither '-A (--all)' or '--ignore-removal',
whose behaviour will change in Git 2.0 with respect to paths you removed.
Paths like 'package.json' that are
removed from your working tree are ignored with this version of Git.

* 'git add --ignore-removal <pathspec>', which is the current default,
  ignores paths you removed from your working tree.

* 'git add --all <pathspec>' will let you also record the removals.

Run 'git status' to check the paths you removed from your working tree.

[adkekiheva@adkekiheva infosec]$ git commit -am "инф-0е3"
[master 1e526ec] инф-0е3
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/placing 800_600_tech.jpg
create mode 100644 labs/lab1/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100755 labs/lab1/report/pandoc/filters/pandoc_eqnos.py
```

Рис. 4.31: Удаление файлов json и создание папок для лабораторных

14. Отправила файлы на сервер (рис. @fig:032).

```
[adkekisheva@adkekisheva infosec]$ git push
warning: push.default is unset; its implicit value is changing in
Git 2.0 from 'matching' to 'simple'. To squelch this message
and maintain the current behavior after the default changes, use:

  git config --global push.default matching

To squelch this message and adopt the new behavior now, use:

  git config --global push.default simple

See 'git help config' and search for 'push.default' for further information.
(the 'simple' mode was introduced in Git 1.7.11. Use the similar mode
'current' instead of 'simple' if you sometimes use older versions of Git)

Counting objects: 35, done.
Compressing objects: 100% (29/29), done.
Writing objects: 100% (34/34), 342.07 KiB | 0 bytes/s, done.
Total 34 (delta 4), reused 0 (delta 0)
remote: Resolving deltas: 100% (4/4), completed with 1 local object.
remote: To git@github.com:adkekisheva/study_2023-2024_infosec.git
11fa481..1e52eec master -> master
```

Рис. 4.32: Отправка файлов в репозиторий

15. После выполнения отчёта преобразовала файлы из Markdown в pdf и docx.

5 Выводы

- Приобрела практических навыки установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.
- Поработала с git, вспомнила работу с ним, установила программное обеспечение.

6 Список литературы

1. Пособие по установке виртуальной машины
2. Задания к лабораторным
3. Администрирование систем Linux. Монтирование файловых систем