

Отчет по лабораторной работе №1

Операционные системы

Самарханова Полина Тимуровна

Содержание

1 Цель работы

Целью данной работы является приобретение практических навыков установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.

2 Задания

1. Создание виртуальной машины
2. Установка операционной системы
3. Работа с операционной системой после установки
4. Установка ПО для создания документации
5. Дополнительные задания

3 Выполнение лабораторной работы

3.1 Создание виртуальной машины

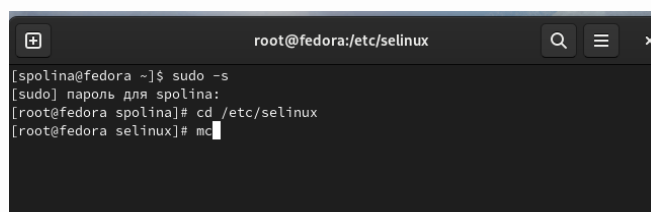
VirtualBox был установлен мной еще в первом семестре в курсе “Архитектура компьютера”

3.2 Установка операционной системы

Также в первом семестре мной уже была установлена и полностью настроена Fedora

3.3 Работа с операционной системой после установки

Все настройки были проведены и удовлетворят требованиям лабораторной работы, за исключением одной: нужно было отключить систему безопасности Selinux. Для этого я открыла консоль и прописала следующие команды: `sudo -s` -она нужна для того, чтобы получить права супер-пользователя затем я перемещаюсь в директорию `/etc/selinux` и открываю `mc` (рис. 3.1)



```
root@fedora:/etc/selinux
[spolina@fedora ~]$ sudo -s
[sudo] пароль для spolina:
[root@fedora spolina]# cd /etc/selinux
[root@fedora selinux]# mc
```

Рис. 3.1 Поиск файла настроек selinux

В открывшемся окне нахожу и открываю файл `config` (рис. 3.2)

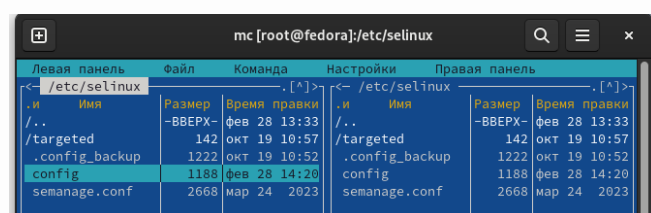
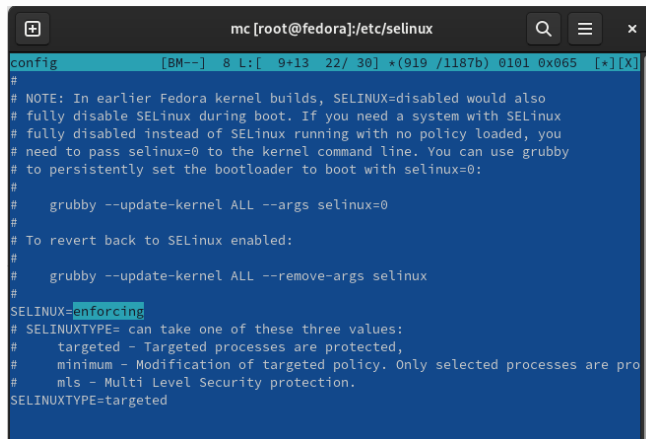


Рис. 3.2 Config file

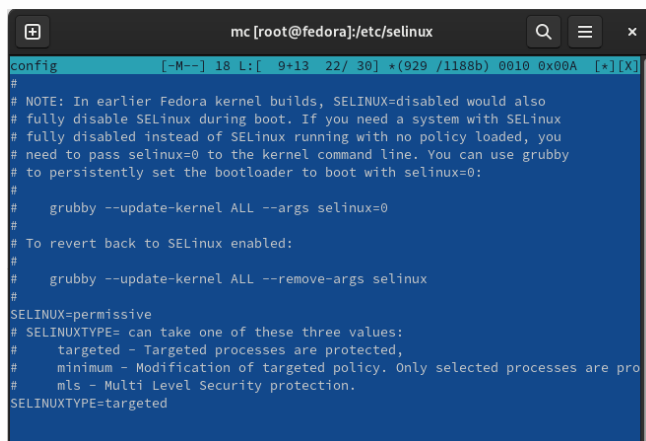
В этом файле нахожу строчку SELINUX=enforcing (рис. 3.3)



```
mc [root@fedora]/etc/selinux
config [BM--] 8 L:[ 9+13 22/ 30] *(919 /1187b) 0101 0x065 [*][X]
#
# NOTE: In earlier Fedora kernel builds, SELINUX=disabled would also
# fully disable SELinux during boot. If you need a system with SELinux
# fully disabled instead of SELinux running with no policy loaded, you
# need to pass selinux=0 to the kernel command line. You can use grubby
# to persistently set the bootloader to boot with selinux=0:
#
# grubby --update-kernel ALL --args selinux=0
#
# To revert back to SELinux enabled:
#
# grubby --update-kernel ALL --remove-args selinux
#
SELINUX=enforcing
# SELINUXTYPE= can take one of these three values:
#   targeted - Targeted processes are protected,
#   minimum - Modification of targeted policy. Only selected processes are pro
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

Рис. 3.3 Параметр Selinux

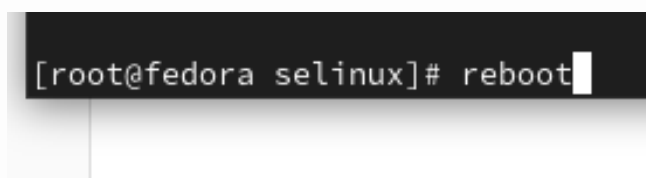
Далее меняю параметр enforcing на permissive и закрываю окно с сохранением (рис. 3.4)



```
mc [root@fedora]/etc/selinux
config [-M--] 18 L:[ 9+13 22/ 30] *(929 /1188b) 0010 0x00A [*][X]
#
# NOTE: In earlier Fedora kernel builds, SELINUX=disabled would also
# fully disable SELinux during boot. If you need a system with SELinux
# fully disabled instead of SELinux running with no policy loaded, you
# need to pass selinux=0 to the kernel command line. You can use grubby
# to persistently set the bootloader to boot with selinux=0:
#
# grubby --update-kernel ALL --args selinux=0
#
# To revert back to SELinux enabled:
#
# grubby --update-kernel ALL --remove-args selinux
#
SELINUX=permissive
# SELINUXTYPE= can take one of these three values:
#   targeted - Targeted processes are protected,
#   minimum - Modification of targeted policy. Only selected processes are pro
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

Рис. 3.4 Изменяю значение Selinux с enforcing на permissive

После этого перезагружаю систему с помощью команды reboot (рис. 3.5)



```
[root@fedora selinux]# reboot
```

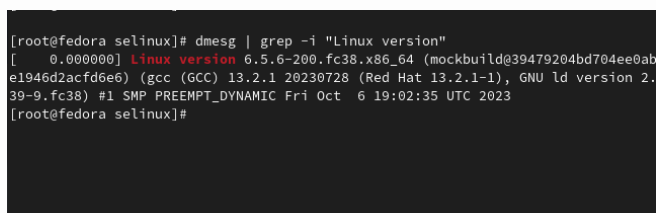
Рис. 3.5 Перезагрузка системы

3.4 Установка ПО для создания документации

Все необходимое ПО (Pandoc, Pandoc-crossref, TexLive) было установлено еще при прохождении раздела “Архитектура компьютера”

#Домашние задания С помощью данной команды `dmesg | grep -i` “то, что ищем” нужно было получить следующую информацию о системе и конфигурации компьютера: 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. Последовательность монтирования файловых систем.

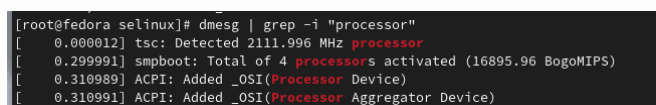
С помощью команды `dmesg | grep -i “Linux version”` ищу версию ядра Linux



```
[root@fedora selinux]# dmesg | grep -i "Linux version"
[ 0.000000] Linux version 6.5.6-200.fc38.x86_64 (mockbuild@39479204bd704ee0abe1946d2acfd6e6) (gcc (GCC) 13.2.1 20230728 (Red Hat 13.2.1-1), GNU ld version 2.39-9.fc38) #1 SMP PREEMPT_DYNAMIC Fri Oct 6 19:02:35 UTC 2023
[root@fedora selinux]#
```

Рис. 3.6 Поиск версии ядра

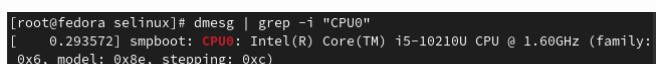
Аналогично ищу частоту процессора, используя команду `dmesg | grep -i “processor”`



```
[root@fedora selinux]# dmesg | grep -i "processor"
[ 0.000012] tsc: Detected 2111.996 MHz processor
[ 0.299991] smpboot: Total of 4 processors activated (16895.96 BogoMIPS)
[ 0.310989] ACPI: Added _OSI(Processor Device)
[ 0.310991] ACPI: Added _OSI(Processor Aggregator Device)
```

Рис. 3.7 Частота процессора

Далее нахожу название модели процессора



```
[root@fedora selinux]# dmesg | grep -i "CPU0"
[ 0.293572] smpboot: CPU0: Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz (family: 0x6, model: 0x8e, stepping: 0xc)
```

Рис. 3.8 Модель процессора

Потом нужно было найти объем доступной оперативной памяти, работаю аналогично

```
[root@fedora selinux]# dmesg | grep -i "memory"
[ 0.002943] ACPI: Reserving FACP table memory at [mem 0xdfff00f0-0xdfff01e3]
[ 0.002945] ACPI: Reserving DSDT table memory at [mem 0xdfff0480-0xdfff27a4]
[ 0.002946] ACPI: Reserving FACS table memory at [mem 0xdfff0200-0xdfff023f]
[ 0.002947] ACPI: Reserving FACS table memory at [mem 0xdfff0200-0xdfff023f]
[ 0.002949] ACPI: Reserving APIC table memory at [mem 0xdfff0240-0xdfff02ab]
[ 0.002950] ACPI: Reserving SSDT table memory at [mem 0xdfff02b0-0xdfff047b]
[ 0.003558] Early memory node ranges
[ 0.021012] PM: hibernation: Registered nosave memory: [mem 0x00000000-0x00000fff]
[ 0.021015] PM: hibernation: Registered nosave memory: [mem 0x0009f000-0x0009ffff]
[ 0.021016] PM: hibernation: Registered nosave memory: [mem 0x000a0000-0x000effff]
[ 0.021017] PM: hibernation: Registered nosave memory: [mem 0x000f0000-0x000fffff]
```

Рис. 3.9 Доступная оперативная память

Нахожу тип обнаруженного гипервизора, используя команду `dmesg | grep -i "hypervisor"`

```
[root@fedora selinux]# dmesg | grep -i "hypervisor"
[ 0.000000] Hypervisor detected: KVM
[ 0.121860] SRBDS: Unknown: Dependent on hypervisor status
[ 0.121862] GDS: Unknown: Dependent on hypervisor status
[ 4.791086] vmwgfx 0000:00:02.0: [drm] *ERROR* vmwgfx seems to be running on an unsupported hypervisor.
[root@fedora selinux]#
```

Рис. 3.10 Поиск типа обнаруженного гипервизора

Тип файловой системы корневого раздела пришлось искать с помощью другой команды: `df -Th | grep -i "^/dev"`

```
[root@fedora selinux]# df -Th | grep -i "^/dev"
/dev/sda3 btrfs 34G 15G 20G 43% /
/dev/sda3 btrfs 34G 15G 20G 43% /home
/dev/sda2 ext4 974M 258M 649M 29% /boot
[root@fedora selinux]#
```

Рис. 3.11 Поиск типа файловой системы корневого раздела

Последовательность монтирования файловых систем можно найти с помощью команды `dmesg | grep -i "mounted"`

```
[root@fedora selinux]# dmesg | grep -i "mounted"
[ 9.327422] systemd[1]: Mounted dev-hugepages.mount - Huge Pages File System.
[ 9.328586] systemd[1]: Mounted dev-mqueue.mount - POSIX Message Queue File System.
[ 9.329774] systemd[1]: Mounted sys-kernel-debug.mount - Kernel Debug File System.
[ 9.333025] systemd[1]: Mounted sys-kernel-tracing.mount - Kernel Trace File System.
[ 11.074331] EXT4-fs (sda2): mounted filesystem 70ab2480-743b-44eb-9d43-8d58ba43da58 r/w with ordered data mode. Quota mode: none.
[root@fedora selinux]#
```

Рис. 3.12 Поиск типа файловой системы корневого раздела

4 Выводы

При выполнении данной лабораторной работы я приобрела практические навыки установки операционной системы на виртуальную машину, а также сделала настройки минимально необходимых для дальнейшей работы сервисов

5 Ответы на контрольные вопросы

1. Учетная запись содержит необходимые для идентификации пользователя при подключении к системе данные, а так же информацию для авторизации и учета: системного имени (user name) (оно может содержать только латинские буквы и знак нижнее подчеркивание, еще оно должно быть уникальным), идентификатор пользователя (UID) (уникальный идентификатор пользователя в системе, целое положительное число), идентификатор группы (CID) (группа, к к-рой относится пользователь. Она, как минимум, одна, по умолчанию - одна), полное имя (full name) (Могут быть ФИО), домашний каталог (home directory) (каталог, в к-рый попадает пользователь после входа в систему и в к-ром хранятся его данные), начальная оболочка (login shell) (командная оболочка, к-рая запускается при входе в систему).
2. Для получения справки по команде: <команда> -help; для перемещения по файловой системе - cd; для просмотра содержимого каталога - ls; для определения объёма каталога - du <имя каталога>; для создания / удаления каталогов - mkdir/rmdir; для создания / удаления файлов - touch/rm; для задания определённых прав на файл / каталог - chmod; для просмотра истории команд - history
3. Файловая система - это порядок, определяющий способ организации и хранения и именования данных на различных носителях информации. Примеры: FAT32 представляет собой пространство, разделенное на три части: одна область для служебных

структур, форма указателей в виде таблиц и зона для хранения самих файлов. ext3/ext4 - журналируемая файловая система, используемая в основном в ОС с ядром Linux.

4. С помощью команды `df`, введя ее в терминале. Это утилита, которая показывает список всех файловых систем по именам устройств, сообщает их размер и данные о памяти. Также посмотреть подмонтированные файловые системы можно с помощью утилиты `mount`.
5. Чтобы удалить зависший процесс, вначале мы должны узнать, какой у него `id`: используем команду `ps`. Далее в терминале вводим команду `kill < id процесса >`. Или можно использовать утилиту `killall`, что “убьет” все процессы, которые есть в данный момент, для этого не нужно знать `id` процесса.

Список литературы

1. Dash, P. Getting Started with Oracle VM VirtualBox / P. Dash. – Packt Publishing Ltd, 2013. – 86 сс.
2. Colvin, H. VirtualBox: An Ultimate Guide Book on Virtualization with VirtualBox. VirtualBox / H. Colvin. – CreateSpace Independent Publishing Platform, 2015. – 70 сс.
3. Vugt, S. van. Red Hat RHCSA/RHCE 7 cert guide : Red Hat Enterprise Linux 7 (EX200 and EX300) : Certification Guide. Red Hat RHCSA/RHCE 7 cert guide / S. van Vugt. – Pearson IT Certification, 2016. – 1008 сс.
4. Робачевский, А. Операционная система UNIX / А. Робачевский, С. Немнюгин, О. Стесик. – 2-е изд. – Санкт-Петербург : БХВ-Петербург, 2010. – 656 сс.
5. Немет, Э. Unix и Linux: руководство системного администратора. Unix и Linux / Э. Немет, Г. Снайдер, Т.Р. Хейн, Б. Уэйли. – 4-е изд. – Вильямс, 2014. – 1312 сс.
6. Колисниченко, Д.Н. Самоучитель системного администратора Linux : Системный администратор / Д.Н. Колисниченко. – Санкт-Петербург : БХВ-Петербург, 2011. – 544 сс.
7. Robbins, A. Bash Pocket Reference / A. Robbins. – O'Reilly Media, 2016. – 156 сс.