Отчёт по лабораторной работе №6

Дисциплина: Основы администрирования операционных систем

Бызова Мария Олеговна

Содержание

# 1 Цель работы

Целью данной работы является получение навыков управления процессами операционной системы.

# 2 Задание

1. Продемонстрируйте навыки управления заданиями операционной системы (см. раз- дел 6.4.1).
2. Продемонстрируйте навыки управления процессами операционной системы (см. раз- дел 6.4.2).
3. Выполните задания для самостоятельной работы (см. раздел 6.5)

# 3 Выполнение лабораторной работы

## 3.1 Управление заданиями

Для начала получим полномочия администратора su – и введём следующие команды:

sleep 3600 & dd if=/dev/zero of=/dev/null & sleep 7200

Поскольку мы запустили последнюю команду без & после неё, у нас есть 2 часа, прежде чем мы снова получим контроль над оболочкой. Введём Ctrl + z , чтобы остановить процесс. Затем введём jobs и увидим три задания, которые мы только что запустили. Первые два имеют состояние Running, а последнее задание в настоящее время находится в состоянии Stopped. Для продолжения выполнения задания 3 в фоновом режиме введём bg 3 и с помощью команды jobs посмотрим изменения в статусе заданий (рис. 1).

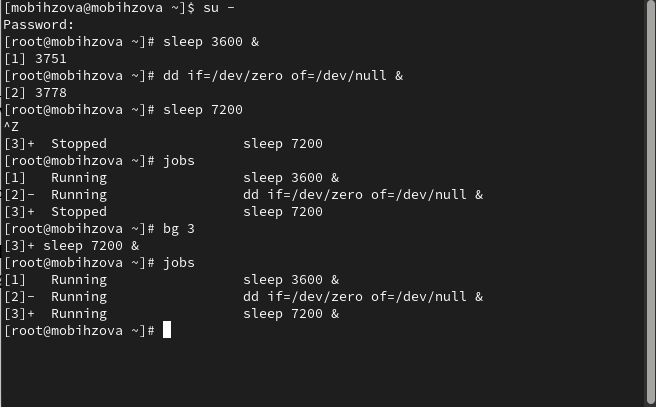


Рис. 1: Получение полномочий администратора, ввод трёх команд, остановка процесса, установка выполнения задания 3 в фоновом режиме, просмотр изменений в статусе заданий

Для перемещения задания 1 на передний план введём fg 1, далее введём Ctrl+ c, чтобы отменить задание 1. С помощью команды jobs посмотрим изменения в статусе заданий и проделаем то же самое для отмены заданий 2 и 3 (рис. 2).

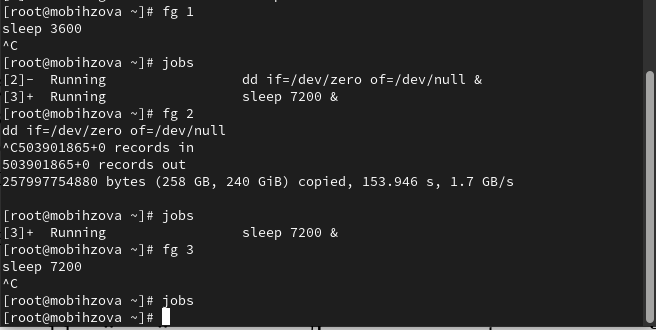


Рис. 2: Перемещение заданий на передний план и их последующая отмена.

Теперь откроем второй терминал и под учётной записью пользователя введём в нём: dd if=/dev/zero of=/dev/null &. После введём exit, чтобы закрыть второй терминал (рис. 3).

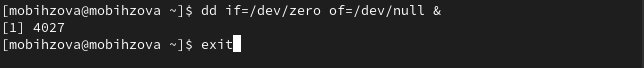


Рис. 3: Ввод команды и закрытие терминала.

На другом терминале под учётной записью своего пользователя запустим top. Мы увидим, что задание dd всё ещё запущено. Для выхода из top используем q и вновь запусткаем top, в нём используем k, чтобы убить задание dd. После этого выйдем из top (рис. 4).

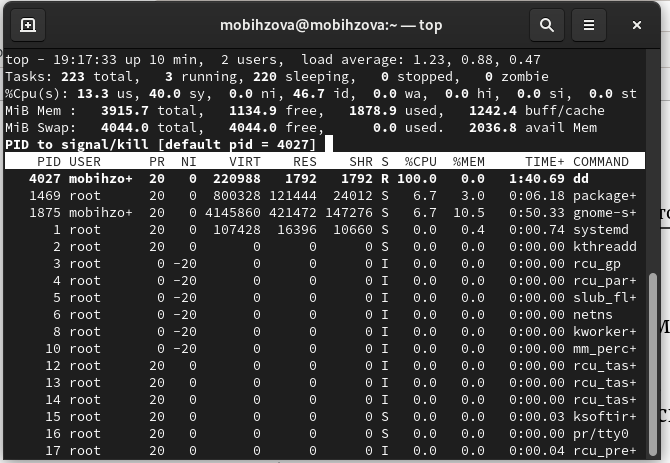


Рис. 4: Убийство задания dd в top.

## 3.2 Управление процессами

Получим полномочия администратора su - и введём следующие команды:

dd if=/dev/zero of=/dev/null & dd if=/dev/zero of=/dev/null & dd if=/dev/zero of=/dev/null &

После чего введём ps aux | grep dd, которое показывает все строки, в которых есть буквы dd. Запущенные процессы dd идут последними. Используем PID первого процесса dd, чтобы изменить приоритет (renice -n 5) (рис. 5).

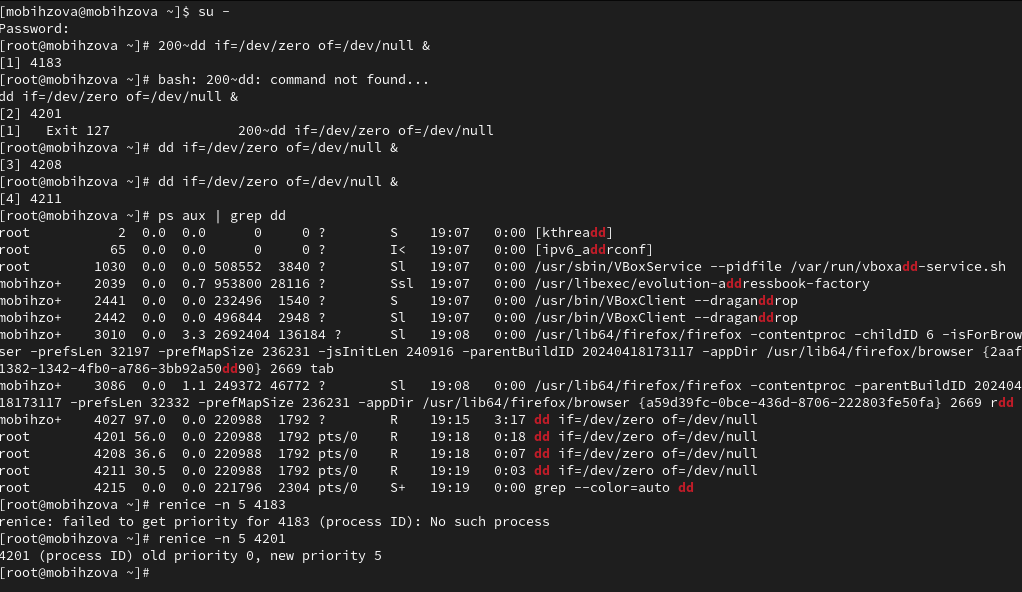


Рис. 5: Получение полномочий администратора, ввод команд. Просмотр всех строк, в которых есть dd. Изменение приоритета.

Введём ps fax | grep -B5 dd. Параметр -B5 показывает соответствующие запросу строки, включая пять строк до этого. Поскольку ps fax показывает иерархию отношений между процессами, мы также видим оболочку, из которой были запущены все процессы dd, и её PID (рис. 6).

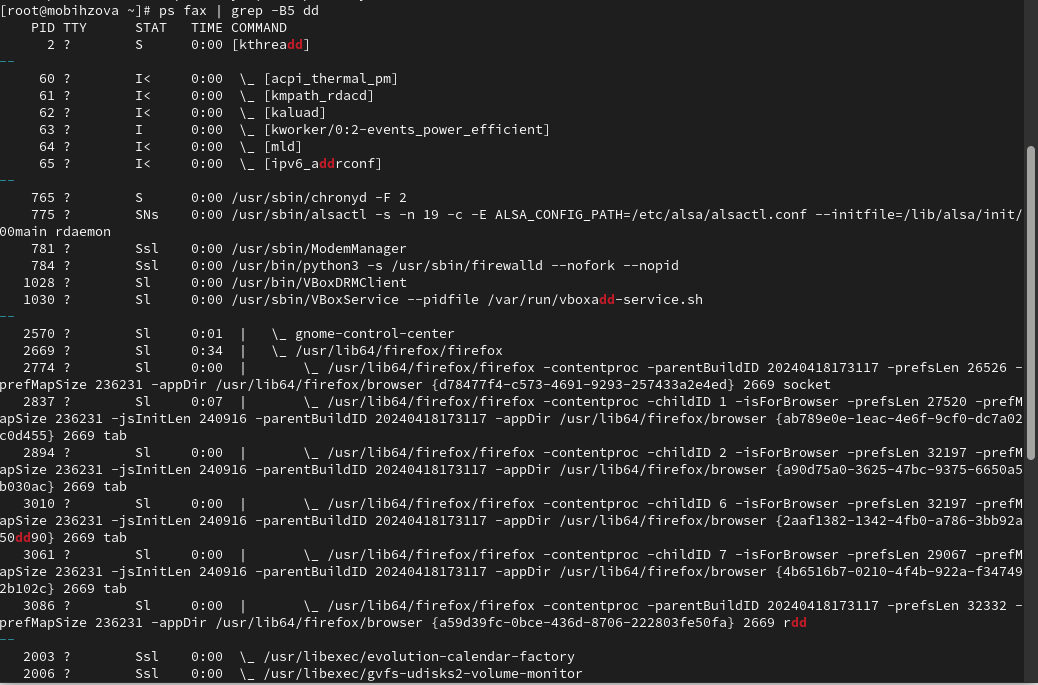


Рис. 6: Просмотр иерархии отношений между процессами.

Теперь найдём PID корневой оболочки, из которой были запущены процессы dd, и введём kill -9 (указав PID оболочки). Мы увидим, что наша корневая оболочка закрылась, а вместе с ней и все процессы dd (остановка родительского процесса — простой и удобный способ остановить все его дочерние процессы) (рис. 7).

Закрытие корневой оболочки.

Рис. 7: Закрытие корневой оболочки.

# 4 Выполнение заданий для самостоятельной работы

## 4.1 Самостоятельная работа (задание 1)

Получим полномочия администратора su – и запустим команду dd if=/dev/zero of=/dev/null & трижды как фоновое задание. Затем увеличим приоритет первой команды, используя значение приоритета −5, после чего изменим приоритет того же процесса ещё раз, но используем на этот раз значение −15 (мы можем менять приоритет команды от -20 (самый высокий приоритет) до 19 (самый низкий приоритет)). Завершим все процессы dd, которые мы запустили командой: killall dd (рис. 8, 9, 10, 11).

Получение полномочий администратора, запуск команды трижды как фоновое задание.

Рис. 8: Получение полномочий администратора, запуск команды трижды как фоновое задание.

Увеличение приоритета первой команды.

Рис. 9: Увеличение приоритета первой команды.

Увеличение приоритета первой команды.

Рис. 10: Увеличение приоритета первой команды.

Завершение всех процессов.

Рис. 11: Завершение всех процессов.

## 4.2 Самостоятельная работа (задание 2)

Получим полномочия администратора su – и запустим программу yes в фоновом режиме с подавлением потока вывода (yes > /dev/null &), далее запустим программу yes на переднем плане с подавлением потока вывода и приостановим выполнение программы. Заново запустим программу yes с теми же параметрами, затем завершим её выполнение. Повторим действия, но уже запустим программу yes на переднем плане без подавления потока вывода (yes > /dev/null). Также приостановим выполнение программы и заново запустим программу yes с теми же параметрами, затем завершим её выполнение. Проверим состояния заданий, воспользовавшись командой jobs. Далее переведём процесс, который у нас выполняется в фоновом режиме, на передний план, затем остановим его (fg 1, после чего Ctrl+c). Переведём 3 процесс с подавлением потока вывода в фоновый режим (bg 3) и проверим состояния заданий, воспользовавшись командой jobs. Обратим внимание, что процесс стал выполняющимся (Running) в фоновом режиме. Запустим процесс в фоновом режиме таким образом, чтобы он продолжил свою работу даже после отключения от терминала (nohup yes > /dev/null &). Закроем окно и заново запустим консоль. Убедимся, что процесс продолжил свою работу (рис. 12).

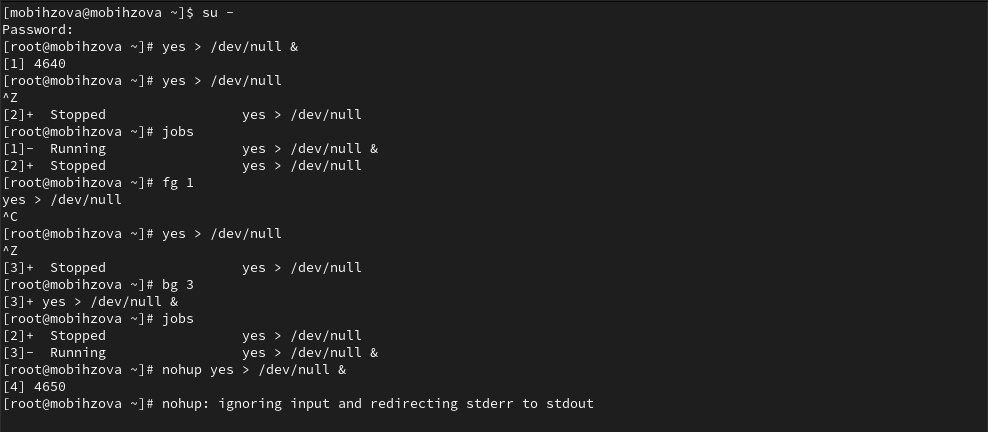


Рис. 12: Получение полномочий администратора. Запуск программы yes в фоновом режиме с подавлением потока вывода. Запуск программы yes на переднем плане без подавления потока вывода. Перевод процесса на передний план и его остановка. Перевод процесса в фоновый режим. Проверка состояния заданий. Запуск процесса в фоновом режиме с условиями.

Сейчас получим информацию о запущенных в операционной системе процессах с помощью утилиты top (рис. 13).

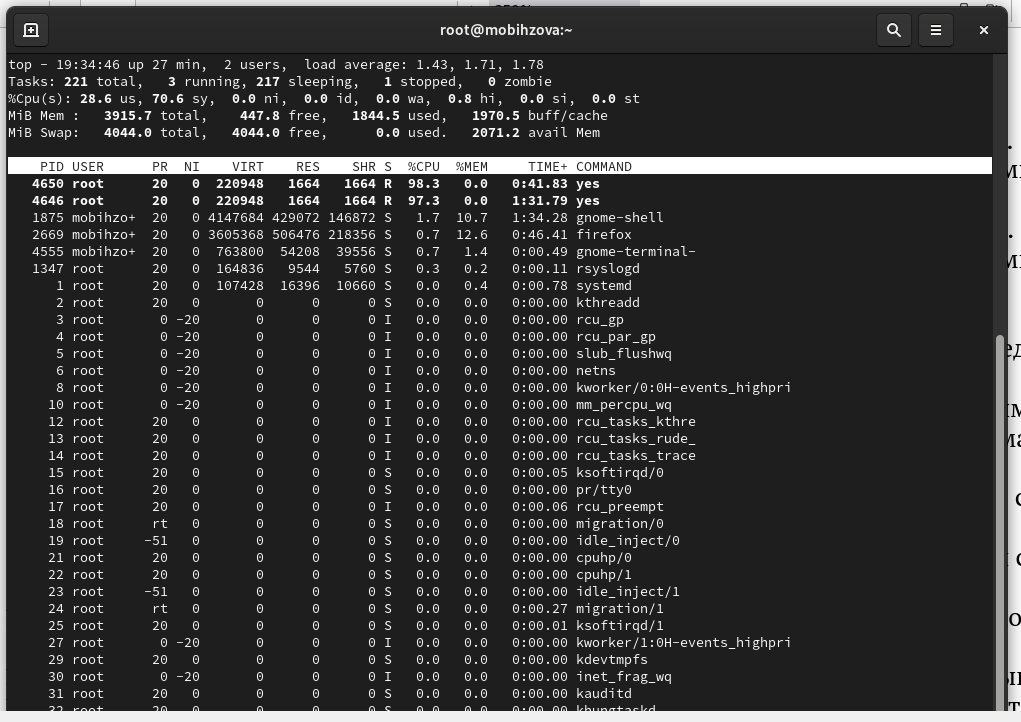


Рис. 13: Получение информации о запущенных в операционной системе процессах.

Запустим ещё три программы yes в фоновом режиме с подавлением потока вывода (yes > /dev/null &). Убьём два процесса: для одного используем его PID (kill -9), а для другого — его идентификатор конкретного задания (fg 2 и Ctrl+c). Попробуем послать сигнал 1 (SIGHUP) процессу, запущенному с помощью nohup (kill -1), и обычному процессу (kill -1) (рис. 14).

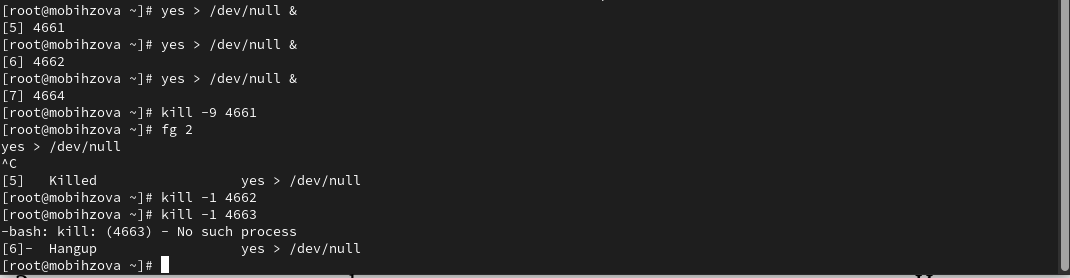


Рис. 14: Запуск трёх программ yes в фоновом режиме с подавлением потока вывода, убийство двух процессов, попытка послать сигнал 1 (SIGHUP).

Запустим ещё несколько программ yes в фоновом режиме с подавлением потока вывода (yes > /dev/null &) и завершим их работу одновременно, используя команду killall yes (рис. 15).

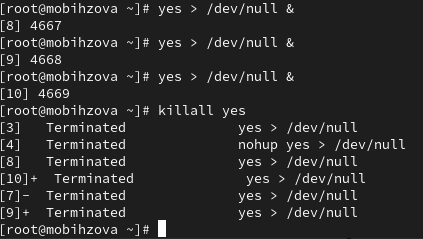


Рис. 15: Запуск программ yes в фоновом режиме с подавлением потока вывода и одновременное завершение их работы.

После чего запустим программу yes в фоновом режиме с подавлением потока вывода (yes > /dev/null &). Используя утилиту nice (nice -n 15 yes), запустим программу yes с теми же параметрами и с приоритетом, большим на 5. Сравним абсолютные и относительные приоритеты у этих двух процессов (ps -l | grep yes). Используя утилиту renice, изменим приоритет у одного из потоков yes таким образом, чтобы у обоих потоков приоритеты были равны (renice -n 15) (рис. 16).

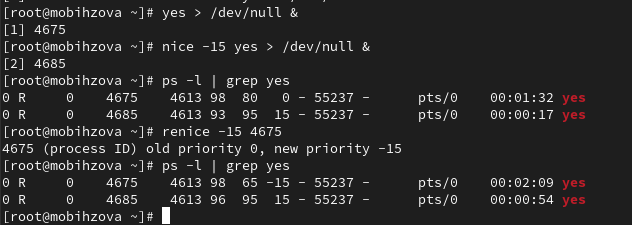
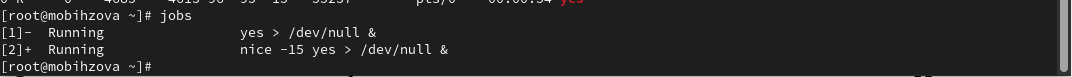


Рис. 16: Запуск программы yes в фоновом режиме с подавлением потока вывода. Запуск программы yes с теми же параметрами и с приоритетом, большим на 5. Сравнение абсолютных и относительных приоритетов, изменение приоритета.

# 5 Ответы на контрольные вопросы

1. Какая команда даёт обзор всех текущих заданий оболочки? jobs.



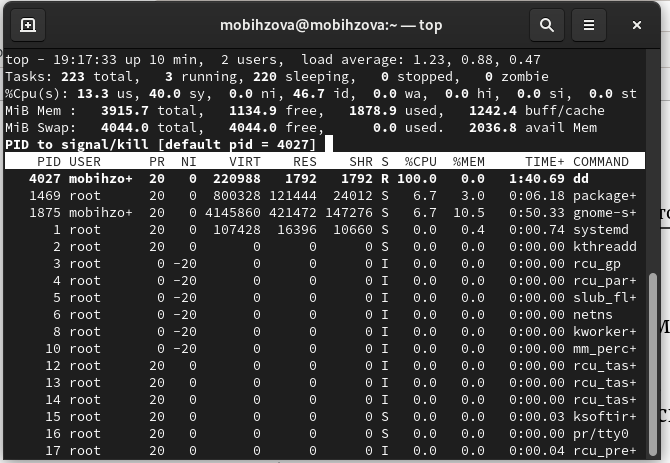
1. Как остановить текущее задание оболочки, чтобы продолжить его выполнение в фоновом режиме? bg номер\_задания



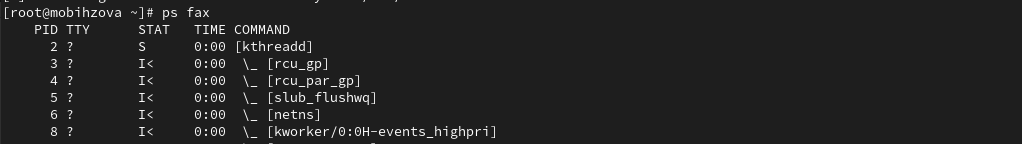
1. Какую комбинацию клавиш можно использовать для отмены текущего задания оболочки? Ctrl+c.



1. Необходимо отменить одно из начатых заданий. Доступ к оболочке, в которой в данный момент работает пользователь, невозможен. Что можно сделать, чтобы отменить задание? Внутри top использовать k, чтобы убить задание.



1. Какая команда используется для отображения отношений между родительскими и дочерними процессами? ps fax.



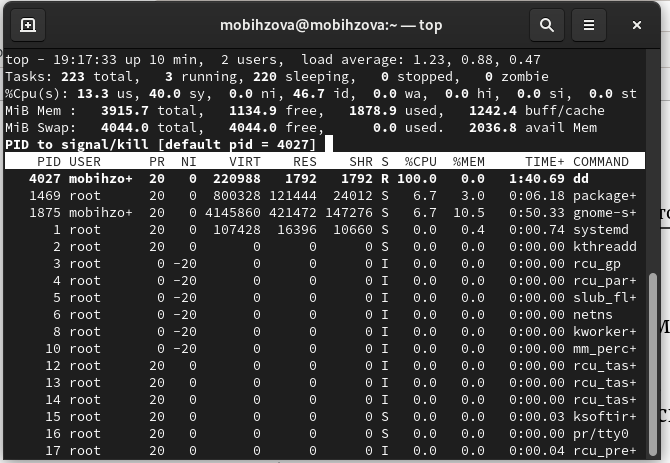
1. Какая команда позволит изменить приоритет процесса с идентификатором 1234 на более высокий? renice -n приоритет\_процесса .



1. В системе в настоящее время запущено 20 процессов dd. Как проще всего остановить их все сразу? killall dd.



1. Какая команда позволяет остановить команду с именем mycommand? Сначала узнаем PID процесса mycommand -ps aux | grep mycommand далее команда kill -9 .
2. Какая команда используется в top, чтобы убить процесс? k.



1. Как запустить команду с достаточно высоким приоритетом, не рискуя, что не хватит ресурсов для других процессов? Запустить команду в фоновом режиме.

# 6 Выводы

В ходе выполнения лабораторной работы были получены навыки управления процессами операционной системы.

# Список литературы

1. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВ-Петербург, 2010.
2. Колисниченко Д. Н. Самоучитель системного администратора Linux. — СПб. : БХВ-Петербург, 2011. — (Системный администратор).
3. Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. — СПб. : Питер, 2015. — (Классика Computer Science).
4. Neil N. J. Learning CentOS: A Beginners Guide to Learning Linux. — CreateSpace Independent Publishing Platform, 2016.
5. Unix и Linux: руководство системного администратора / Э. Немет, Г. Снайдер, Т.Хейн, Б. Уэйли, Д. Макни. — 5-е изд. — СПб. : ООО «Диалектика», 2020.