Лабораторная работа 6

1132232887

Накова Амина Михайловна

Содержание

Список иллюстраций

Список таблиц

# 1 Цель работы

Целью данной работы является получение навыков управления процессами операционной системы.

# 2 Выполнение лабораторной работы

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. 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.

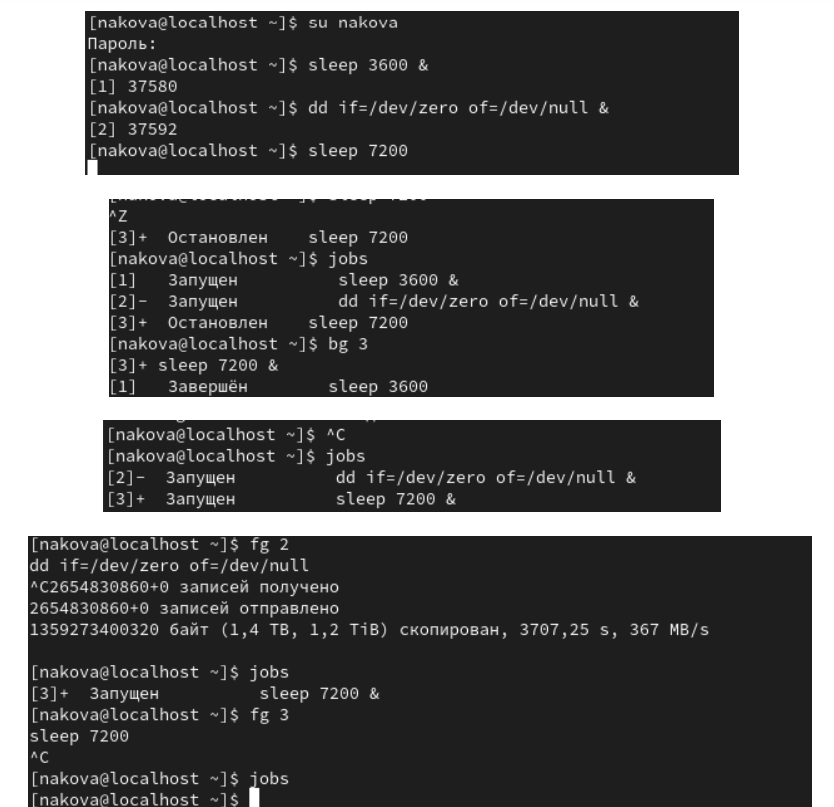


Рис. 1: шаг 1

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.

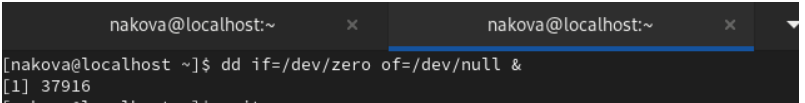


Рис. 2: шаг 2

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

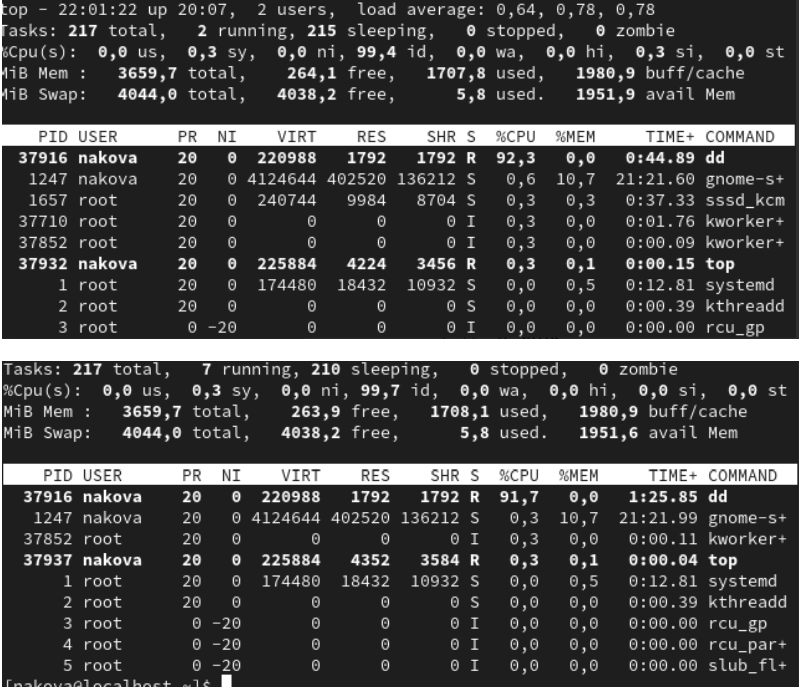


Рис. 3: шаг 3

Управление процессами: Получим полномочия администратора su - и введём следующие команды: 5 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 2682).

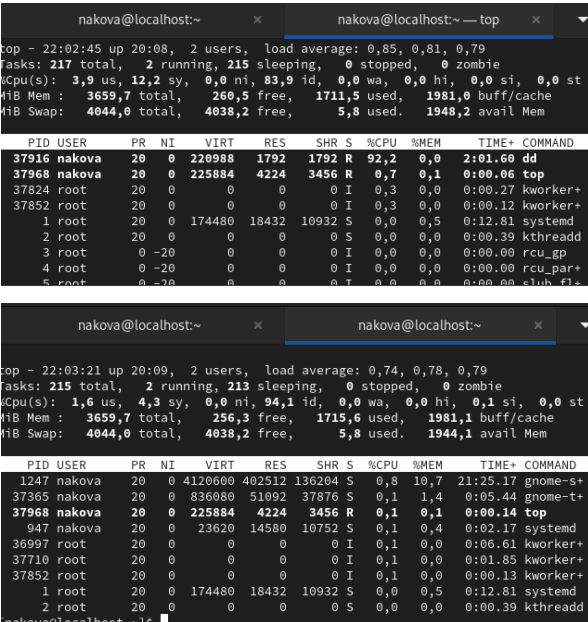


Рис. 4: шаг 4

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. 5: шаг 5

Рис. 5: шаг 5

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 7 процесса — простой и удобный способ остановить все его дочерние процессы).

Рис. 6: шаг 6

Рис. 6: шаг 6

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 8 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

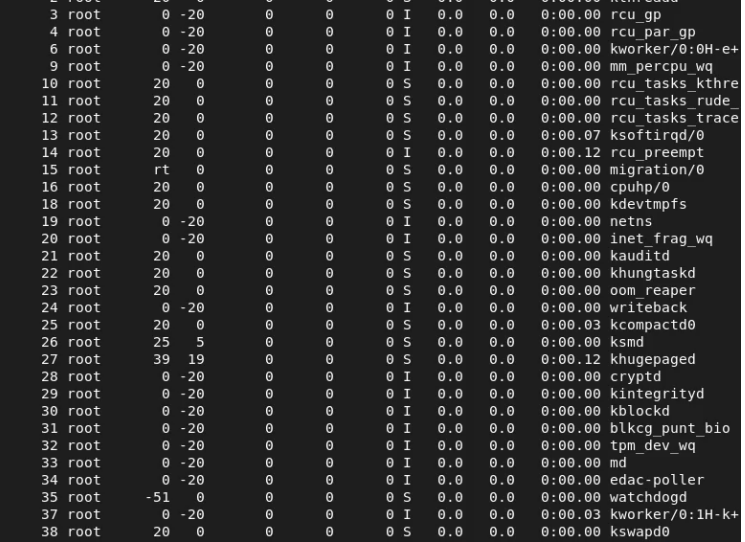


Рис. 7: шаг 7

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. 9 Сейчас получим информацию о запущенных в операционной системе процессах с помощью утилиты top

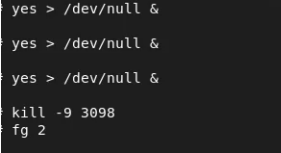


Рис. 8: шаг 8

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 3098), а для другого — его идентификатор конкретного задания (fg 2 и Ctrl+c). Попробуем послать сигнал 1 (SIGHUP) процессу, запущенному с помощью nohup (kill -1 3100), и обычному процессу (kill -1 2993)



Рис. 9: шаг 8

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. 10: шаг 8

Рис. 10: шаг 8

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 3109)

Рис. 11: 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.

Рис. 11: 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.

Ответы на контрольные вопросы: 1. Какая команда даёт обзор всех текущих заданий оболочки? jobs. 2. Как остановить текущее задание оболочки, чтобы продолжить его выполнение в фоновом режиме? bg номер\_задания. 3. Какую комбинацию клавиш можно использовать для отмены текущего задания оболочки? Ctrl+c. 4. 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. 12 5. Какая команда используется для отображения отношений между родительскими и дочерними процессами? ps fax. 6. Какая команда позволит изменить приоритет процесса с идентификатором 1234 на более высокий? renice -n приоритет\_процесса . 7. В системе в настоящее время запущено 20 процессов dd. Как проще всего остановить их все сразу? killall dd. 8. Какая команда позволяет остановить команду с именем mycommand? Сначала узнаем PID процесса mycommand -ps aux | grep mycommand далее команда kill -9 . 9. Какая команда используется в top, чтобы убить процесс? k. 13 10. Как запустить команду с достаточно высоким приоритетом, не рискуя, что не хватит ресурсов для других процессов? Запустить команду в фоновом режиме.

# 3 Выводы

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

# Список литературы