ЛАБОРАТОРНАЯ РАБОТА №5

**на тему: «Возможности и средства операционной системы Linux»**

**Часть 2 — «Диспетчеризация операционной системы Linux»**

по дисциплине «Операционные системы»

Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

(ФГАОУ ВО «ЮФУ»)

Инженерно-технологическая Академия

Институт компьютерных технологий и информационной безопасности

Кафедра Систем Автоматизированного Проектирования

им. В. М. Курейчика

Выполнил

студент КТбо2-4 \_\_\_\_\_\_\_\_\_\_\_\_\_ А. А. Воронов

Принял

доцент каф. САПР, к. т. н. \_\_\_\_\_\_\_\_\_\_\_\_\_ М. Д. Сеченов

Таганрог 2024

Содержание

[Введение 3](#_Toc182401245)

[1 Практическая часть 4](#_Toc182401246)

[1.1 Запуск процесса в фоновом режиме. Просмотр всех процессов, находящихся в фоновом режиме 4](#_Toc182401247)

[1.2 Вывод информации о работающих процессах. Сортировка по параметрам 4](#_Toc182401248)

[1.3 Запись выходного потока процесса в файл 7](#_Toc182401249)

[1.4 Просмотр информации о процессах с помощью системного монитора 7](#_Toc182401250)

[1.5 Вывод информации о работающих процессах. Отправка сигнала на прекращение процесса по ID и по имени 8](#_Toc182401251)

[1.6 Подготовка задания на запуск определенного процесса. Удаление процесса из очереди 9](#_Toc182401252)

[1.7 Запуск процесса с конкретным приоритетом 10](#_Toc182401253)

[1.8 Изменение приоритета конкретного процесса 10](#_Toc182401254)

[1.9 Выяснение уровня выполнения. Переход с текущего уровня выполнения на уровень 1 и обратно 10](#_Toc182401255)

[1.10 Создание at задания для копирования файлов 11](#_Toc182401256)

[1.11 Создание напоминания 12](#_Toc182401257)

[Заключение 14](#_Toc182401258)

[Список использованных источников 15](#_Toc182401259)

Введение

Цель работы — овладеть навыками управления процессами в операционной системе Linux и научиться пользоваться документацией команд. Администрирование процессов играет ключевую роль в обеспечении стабильной работы системы, поскольку позволяет контролировать выполнение программ, оптимизировать ресурсы и устранять потенциальные ошибки.

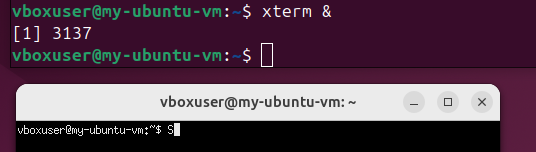
В Linux процессы бывают интерактивными — такими, которые инициируются и управляются в терминальной сессии, и фоновыми, которые не требуют постоянного взаимодействия с пользователем. С помощью оболочки Linux администратор может переключать процессы между передним и фоновым режимами, останавливать их и завершать при необходимости. Утилита «ps» предоставляет пользователю основную информацию о каждом процессе, включая идентификатор (PID), номер родительского процесса (PPID), привязанный терминал и приоритет выполнения.

В работе также будут рассмотрены команды, позволяющие управлять процессами: «jobs» для отображения процессов в фоновом режиме, «kill» для завершения процесса, «nice» для установки приоритетов выполнения, и «pstree» для отображения отношений между процессами.

# Практическая часть

## Запуск процесса в фоновом режиме. Просмотр всех процессов, находящихся в фоновом режиме

Запустил систему. Затем запустил процесс xterm в фоновом режиме (рисунок 1). Далее просмотрел запущенные процессы с помощью команды «jobs» (рисунок 2) [1].



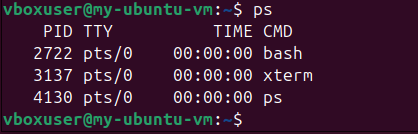
1. — Запуск xterm в фоновом режиме



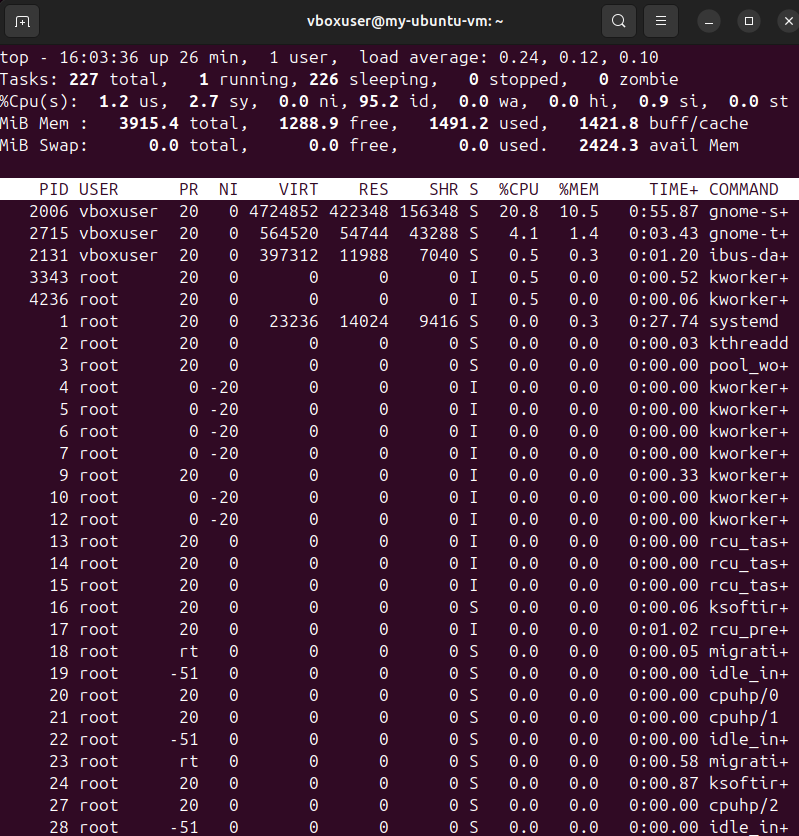
1. — Просмотр запущенных процессов

## Вывод информации о работающих процессах. Сортировка по параметрам

Для отображения информации о работающих процессах можно использовать, например, команды «ps» — чтобы получить краткую информацию (рисунок 3) и «top», чтобы получить полную информацию о всех процессах системы (рисунок 4) [1].

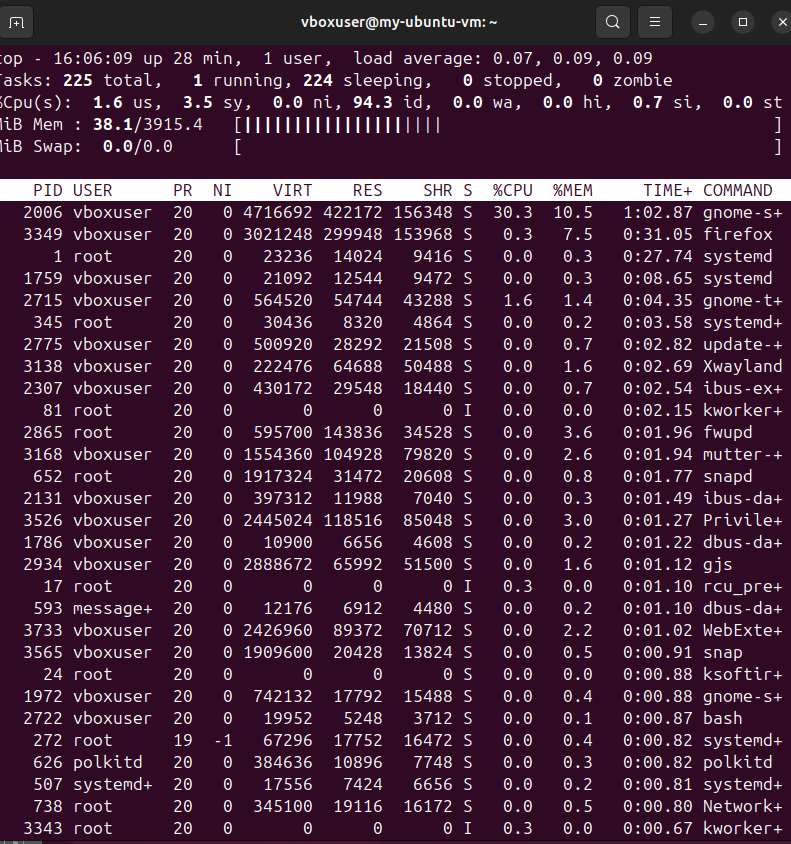


1. — Получение краткой информации о процессах

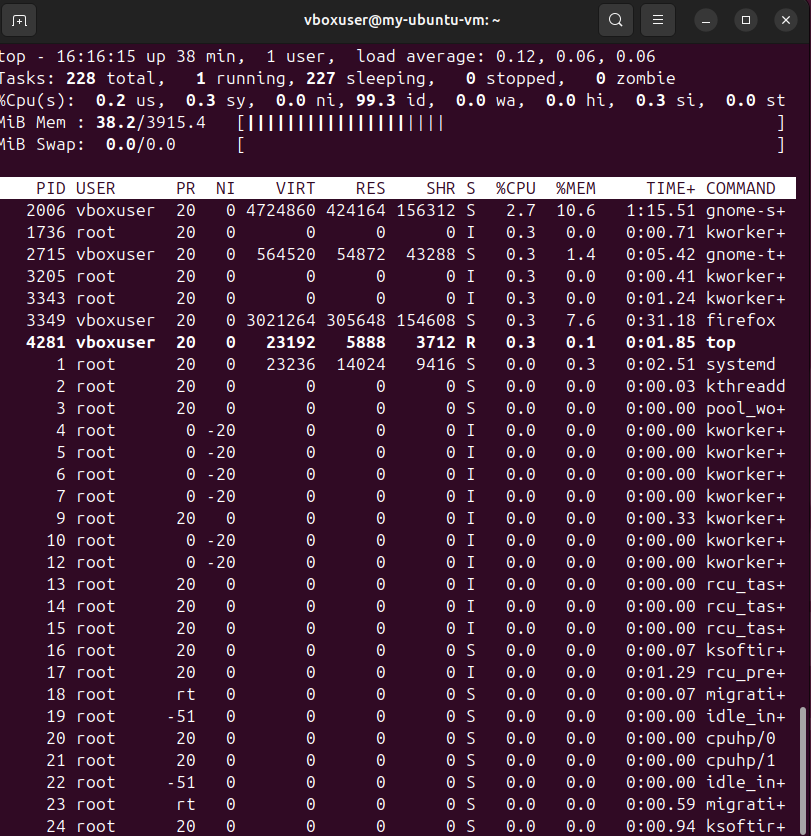


1. — Получение полной информации

Инструмент top позволяет управлять информацией в таблице. Например — сортировать по параметрам времени (рисунок 5) и по используемой процессом памяти (рисунок 6). Для этого после запуска инструмента нужно нажать «T» или «M» соответственно [1]. Инструмент позволяет делать и другие действия.



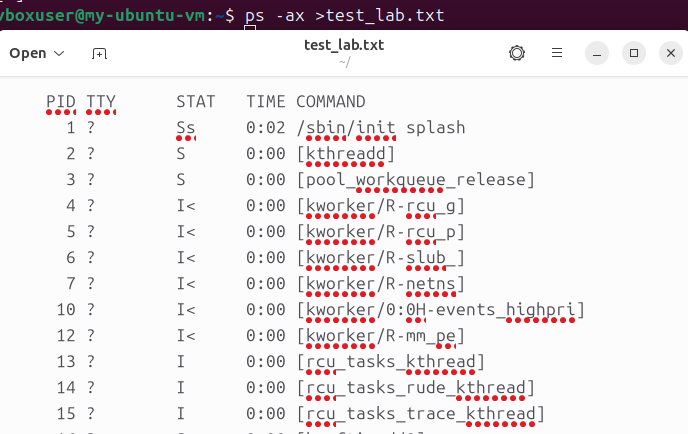
1. — Сортировка по времени



1. — Сортировка по памяти

## Запись выходного потока процесса в файл

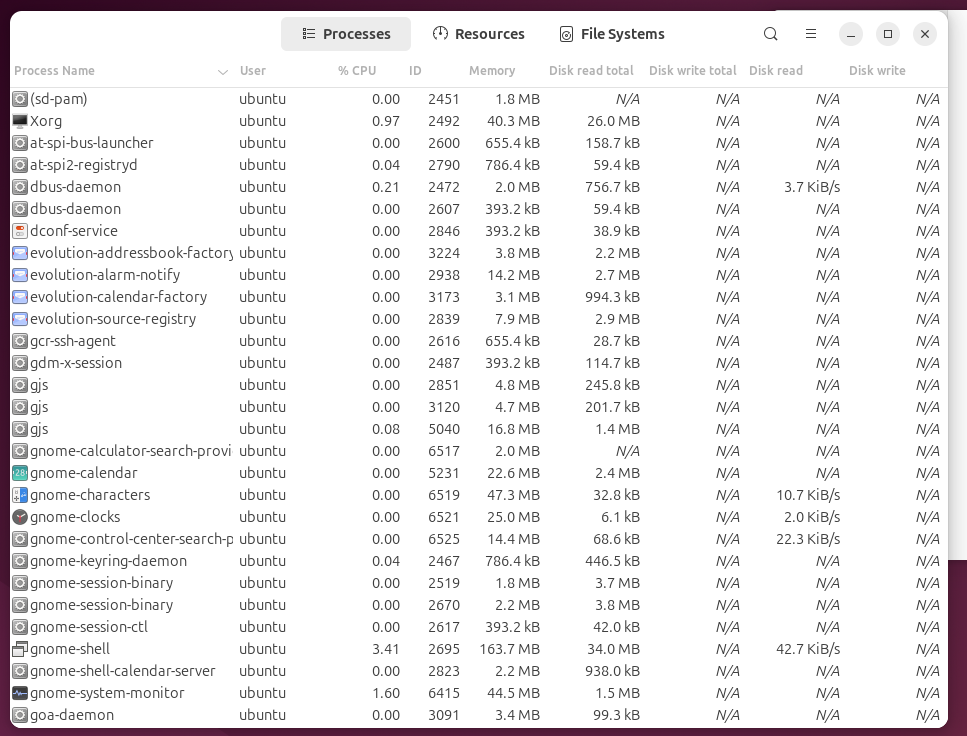
Чтобы записать данные из выходного потока в файл, нужно использовать «ps –ax > <output\_filename> » (рисунок 7).



1. — Копирование выходного потока в файл

## Просмотр информации о процессах с помощью системного монитора

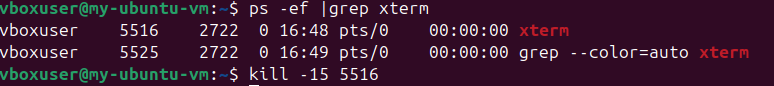
Среди приложений и утилит, установленных в системе, нашел приложение System Monitor и запустил его. В нем можно просмотреть информацию о процессах, файловой системе и ресурсах системы в графическом виде (рисунок 8) [1].



1. — Просмотр информации о процессах в системном мониторе

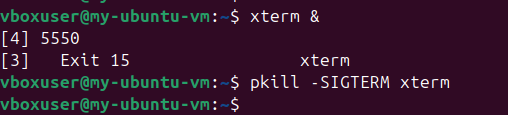
## Вывод информации о работающих процессах. Отправка сигнала на прекращение процесса по ID и по имени

Использовав команду «grep» вывел информацию по конкретному процессу — xterm. Таким образом, я узнал ID процесса и использовал его далее в команде для завершения процесса по айди — «kill -<signal> <proc\_ID>» (рисунок 9) [1]. В данном случае введен код сигнала «15», который соответствует сигналу «TERM» — «завершить процесс».



1. — Завершение процесса по его ID

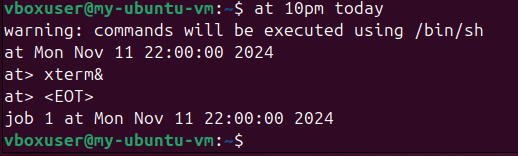
Снова запустил тот же процесс xterm в фоновом режиме, а затем завершил его, обратившись к нему по имени в команде «pkill -<signal> <proc\_name>», передав ему сигнал SIGTERM — завершить процесс «по-тихому» (рисунок 10) [1].



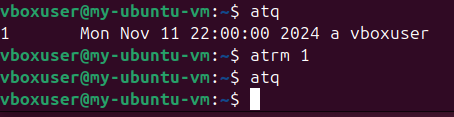
1. — Завершение процесса по его имени

## Подготовка задания на запуск определенного процесса. Удаление процесса из очереди

Запуск команды «at <time>» позволяет создать список команд, которые будут выполнены, когда пройдет установленное время [1]. Я ввел команду, чтобы в 10 часов вечера был запущен процесс xterm в фоновом режиме (рисунок 11). Затем через команду «atq» просмотрел список всех заданий и нашел там ID задания на запуск процесса — «1». Используя выясненное значение, удалил задание из списка командой «atrm <at\_ID>» (рисунок 12).



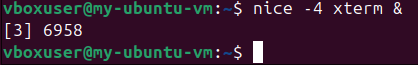
1. — Подготовка задания на запуск процесса



1. — Удаление задания из очереди

## Запуск процесса с конкретным приоритетом

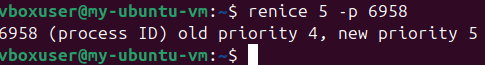
Используя инструмент nice, установил приоритет для конкретного процесса. В данном случае установил приоритет 4 (рисунок 13) [1].



1. — Запуск процесса с конкретным приоритетом

## Изменение приоритета конкретного процесса

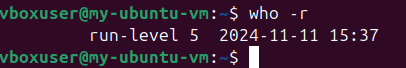
Чтобы переназначить приоритет процессу, использовал команду «renice <priority> -p <proc\_ID>» (рисунок 14).



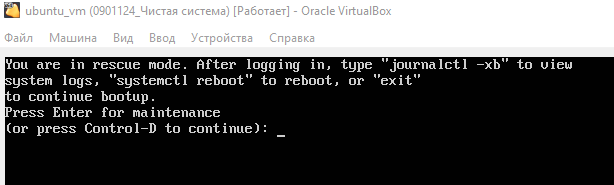
1. — Смена приоритета конкретного процесса

## Выяснение уровня выполнения. Переход с текущего уровня выполнения на уровень 1 и обратно

Выяснил свой уровень выполнения, введя команду «who -r» (рисунок 15) [1]. Текущий уровень равен «5». Далее изменил свой уровень выполнения при помощи команды «telinit» на «1». Система вошла в режим rescue (рисунок 16). Нажав Ctrl+D, запустил систему. Уровень выполнения автоматически стал равен «5».

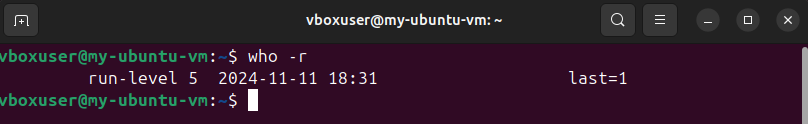


1. — Выяснение текущего уровня выполнения



1. — Вид системы на уровне выполнения «1»

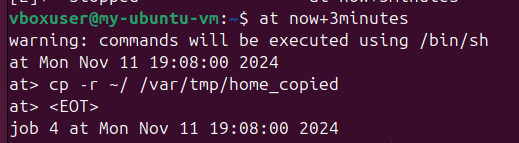
Проверив свой уровень выполнения, убедился, что он автоматически сменился с «1» на «5» (рисунок 17).



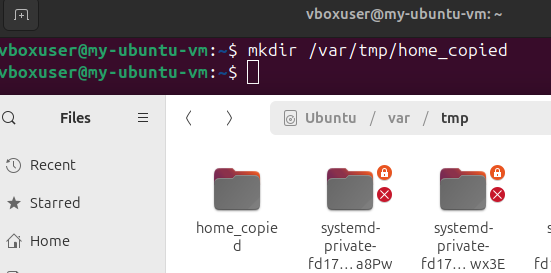
1. — Повторное выяснение уровня выолнения

## Создание at задания для копирования файлов

Используя «at», задал задание по копированию файлов домашнего каталога пользователя в созданную заранее директорию (рисунок 18). Это директория /var/tmp/home\_copied (рисунок 19). Используя «at now+3minutes», создал задание, которое выполнится через 3 минуты после ввода команды. В очередь вставил команду «cp –r ~\ \var\tmp\home\_copied» [1].



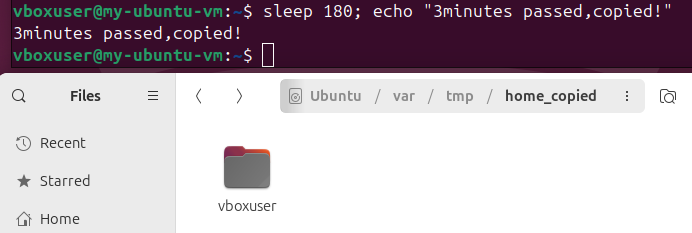
1. — Создание задания по копированию



1. — Создание каталога

## Создание напоминания

Создал напоминание, используя «sleep». В секундах отмерил 3 минуты и указал, чтобы по прошествии времени выполнилась команда «echo», которая выведет в терминал заготовленную символьную строку (рисунок 20). Действительно — через 3 минуты после запуска команды, в терминал вывелась строка, а в процесс копирования из предыдущего пункта успешно завершился.



1. — Создание напоминания

Заключение

В ходе лабораторной работы были получены практические навыки администрирования процессов в операционной системе Linux, включая управление их выполнением, установку приоритетов и завершение задач. Рассмотрены интерактивные и фоновые процессы, их свойства и особенности взаимодействия с терминалом. Навыки управления процессами оказались полезными для повышения стабильности работы системы и рационального использования ресурсов, а также позволили понять ключевые команды и утилиты, используемые для мониторинга и администрирования процессов.

Освоенные инструменты, такие как «ps», «jobs», «kill», «nice» и «pstree», предоставляют возможности для оперативного управления процессами и поддержания оптимальной производительности системы, что является важным аспектом при работе с операционными системами на базе Linux.

Список использованных источников

1. Нужнов, Е. В. \ Методические указания к выполнению комплекса лабораторных работ по общеинститутской дисциплине «Операционные системы» \ Е. В. Нужнов, А. Н. Самойлов, С. Н. Дроздов, М. Д. Сеченов \ [текст] \ Южный федеральный университет. – Таганрог: Издательство Южного федерального университета, 2023. – 145 с.