|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **Компьютерные системы и сети (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.01 Информатика и вычислительная техника

**Отчет**

|  |  |
| --- | --- |
| **по лабораторной работе №** | 4 |

**Название:** Исследование способов организации оперативной памяти и взаимодействия процессов

**Дисциплина:** Операционные системы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-53Б |  |  | В.К. Залыгин |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | В.Ю. Мельников А.М. Суровов |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2024

**ВВЕДЕНИЕ**

**Цель лабораторной работы**

Цель данной работы — получение теоретических и практических сведений об управлении процессами, потоками и оперативной памятью в UNIX-подобных системах и в Linux в частности.

**Задание**

• Открыть в текстовом браузере некую страницу и перевести его в фоновый режим

• Запустить ещё два экземпляра текстового браузера в фоновом режиме

• Найти процесс, максимально нагружающий процессор

• Вывести список процессов текущего пользователя

• «Убить» первый процесс браузера в котором открыта 1 страница

• Вывести список всех процессов всех пользователей

• Просмотреть список процессов постранично

• Отобрать из вывода команды ps строку, соответствующую процессу «dbus-daemon», определить, где лежит её выполняемый файл и с какими параметрами он запущен

• Записать в файл с именем, содержащим текущее время, строку «-------»и список процессов

• Выполнить команду в фоновом режиме с отсрочкой запуска на 1 минуту. Продемонстрировать, что команда выполнилась именно через минуту.

• Отобрать из одного из сформированных файлов строки, относящиеся к одному из процессов.

• Вывести результаты работы произвольной команды в один файл, а сообщения об ошибках в другой. Продемонстрировать правильность работы.

• Выполнить произвольную команду с ограничением использования процессорного времени 300 секунд и выводом результатов и сообщений об ошибках в один файл.

• Настроить cron на выполнение команды ежедневно в заданное время. Продемонстрировать правильность работы.

**Выполнение**

Сначала откроем в консольном браузере сайт и уведем его в фон.



Рисунок 1 – Результат перевода браузера в фоновый режим

Откроем еще две страницы и уведем их в фон.



Рисунок 2 – Еще два запущенных в фоне процесса

Используем утилиту top и посмотрим самый ресурсозатратный процесс.

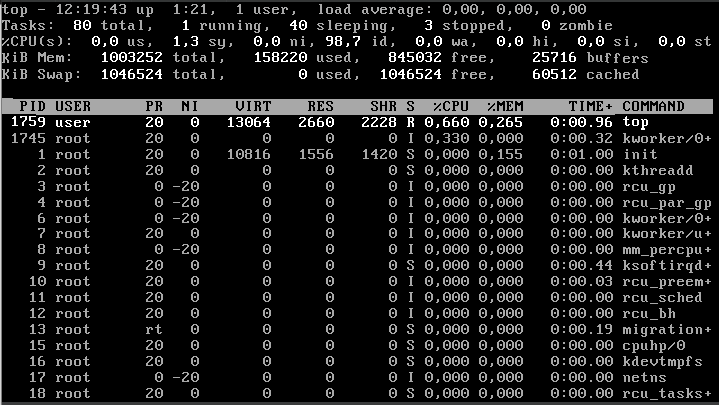


Рисунок 3 – Больше всего ресурсов CPU потребляет программа top

С помощью команды ps a посмотрим все процессы и найдем процессы браузеров. После чего завершим один из них принудительно.

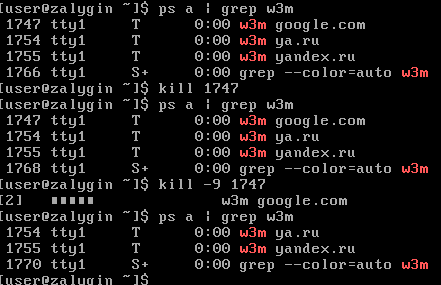


Рисунок 4 – Принудительное завершение процесса первого браузера

Вывод всех процессов пользователей командой ps a.

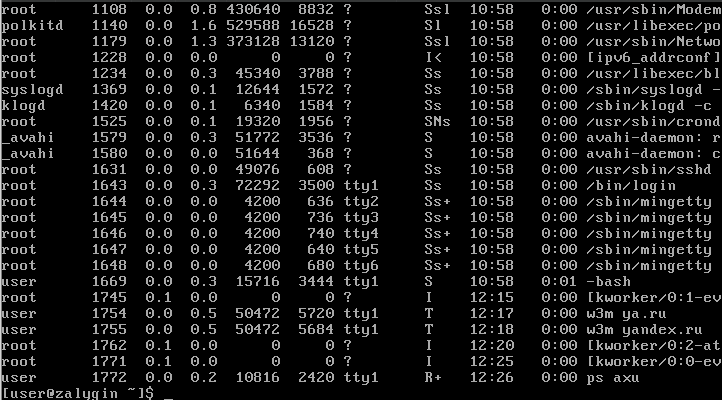


Рисунок 5 – Вывод всех процессов всех пользователей

Так как список выше, чем размер терминала, удобно будет использовать команду последовательной выдачи строк more.

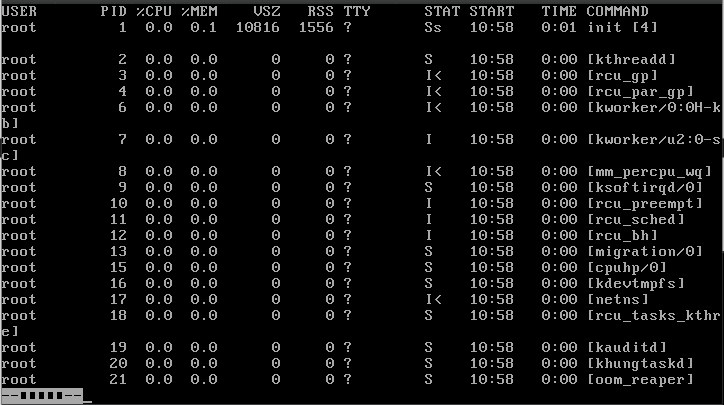


Рисунок 6 – Использование утилиты more для последовательной выдачи данных

Выведем строку, относящуюся к процессу dbus-daemon



Рисунок 7 – Строка вывода ps, соответствующая процессу dbus-daemon

Чтобы создать файл с датой и требуемой строкой, можно использовать группу команд.

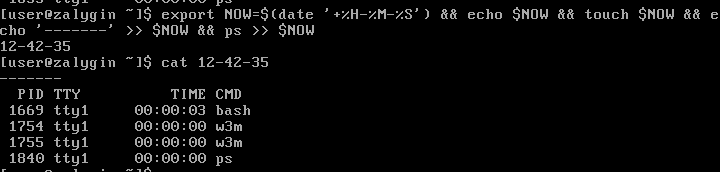


Рисунок 8 – Создание файла с текущем временем, запись в него строки и списка процессов

Чтобы отсрочить команду на минуту, можно ее предварять командой date.

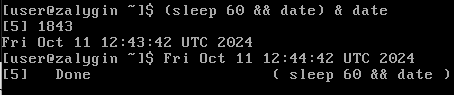


Рисунок 9 – Отсрочка команды на 1 минуту

Используем grep, чтобы вывести строки процессов, относящиеся к браузеру.

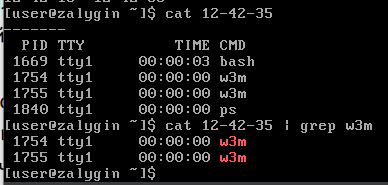


Рисунок 10 – Фильтрация строк, относящихся только к утилите w3m

Чтобы перенаправить потоки вывода и ошибок, нужно использовать специальные операторы >, 2>.

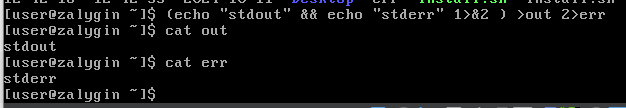


Рисунок 11 – Вывод потока вывода и потока ошибок в раздельные файлы

Для выполнения операции с ограничением по времени работы, можно использовать команду timeout.

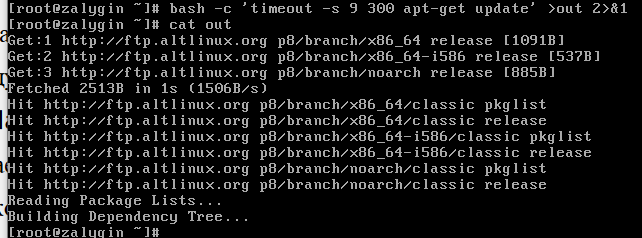


Рисунок 12 – Выполнение операции с ограничением по времени работы

Создаем задание, пишущее в файл текущую дату ежедневно.

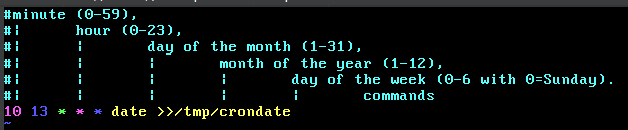


Рисунок 13 – Установка команды с запуском ежедневно в 13:10

После смотрим на содержимое файла.



Рисунок 14 – Результат выполнения задания

**ЗАКЛЮЧЕНИЕ**

В результате выполнения работы были получены теоретические и практические сведения об управлении процессами, потоками и оперативной памятью в UNIX-подобных системах и в Linux в частности.