

# ЛАБОРАТОРНАЯ РАБОТА №7

## УПРАВЛЕНИЕ ПРОЦЕССАМИ В ОС

# УПРАВЛЕНИЕ ПРОЦЕССАМИ В ОС

Цель работы: Научиться управлять процессами ОС. Научиться передавать сигналы процессам. Разобраться в текущем состоянии ОС, насколько система загружена, какие процессы имеют приоритеты. Научиться управлять приоритетами процессов.

Задание:

*Задание (пошагово):*

1. Из консоли пользователя командой **ps-efl | more** выведите расширенный постраничный список исполняемых процессов (перечень параметров для расширенного вывода информации можно уточнить с помощью электронного справочника **manps**).

Разберитесь с выводимой информацией. Определите процессы:

2. по типу: системные, демоны, пользовательские (тип процесса определяется по косвенным признакам, в частности, по имени).

# УПРАВЛЕНИЕ ПРОЦЕССАМИ В ОС

3. по состоянию **S**: (исполняющиеся -**R** или **O**, ожидающие записи на диск -**D**, ожидающие событий -**S**, приостановленные -**T**, зомби -**Z** и т.д.),
4. по текущему динамическому приоритету **PRI** (наименьшее значение у высокоприоритетных процессов),
5. по относительному приоритету **N1**.
6. В консоли суперпользователя запустите утилиту **top** для текущего контроля процессов. Утилита позволяет отобразить наиболее активные процессы (столько, сколько их помещается на экран) с достаточно полной информацией о них (для пользователя утилита представляет ограниченный набор выводимых параметров).
7. Из первой консоли создайте процесс `od/dev/zero > /dev/null`. В соответствии с введенной командой утилита **od** читает и выводит непрерывный поток нулевых байт из «рога изобилия» в нулевое устройство. Переключившись в другую консоль, с помощью команды **top** просмотрите список наиболее активных процессов. Найдите и идентифицируйте запущенный процесс, найдите по идентификатору **PPID** его «родителя», определите его приоритет (возможно это -

# УПРАВЛЕНИЕ ПРОЦЕССАМИ В ОС

величина переменная), долю загрузки центрального процессора **%CPU** и оперативной памяти **%MEM**.

8. Поочередно из первой и второй консолей с правами администратора и пользователя с помощью команды **od/dev/zero> /dev/null&** создайте по 2-3 одинаковых фоновых процесса.
9. По мере создания новых процессов отслеживайте в третьей консоли их текущий приоритет, загрузку процессора и памяти. Имеются ли различия в приоритете процессов, выполняемых от имени администратора и пользователя?
10. С консоли пользователя **user1** измените приоритет одного из принадлежащих ему процессов. Для этого воспользуйтесь командой **renice -10 PID**. Изменился ли относительный приоритет процесса?
11. Повторите предыдущий пункт с правами администратора.
12. Переключитесь в консоль пользователя и измените приоритет одного из принадлежащих ему процессов командой **renice5 PID**. Произошло ли изменение приоритета?

# УПРАВЛЕНИЕ ПРОЦЕССАМИ В ОС

13. Проконтролируйте из третьей консоли изменение приоритетов запущенных процессов.
14. Удалите созданные процессы командой **kill**.
15. Теперь перейдем к управлению процессами. С правами пользователя создайте в своей директории сценарий с именем **abcd**. Сценарий можно создать с помощью команды **cat**:  
**viabcd**  
**#!/bin/bash**  
**while:** remобратите внимание на пробел перед двоеточием!  
**do**  
**echo HELLO!**  
**Done**
16. Используя команду **chmod**, присвойте пользователю полные права на чтение, запись и исполнение данного сценария. Запустите сценарий на исполнение (на экран должны непрерывно выводиться приветствия **HELLO!**)



# УПРАВЛЕНИЕ ПРОЦЕССАМИ В ОС

17. Перейдите в третью консоль, с помощью команды **top** просмотрите список процессов и найдите в нем «зависший» процесс, запущенный пользователем (на самом деле это только имитация зависания, которое пользователь легко может прекратить сам). Прочитайте идентификатор процесса **PID**.

18. Нажатием **Ctrl+C** из второй консоли остановите процесс. Как изменилось при этом состояние процесса?

19. Повторно запустите из второй консоли процесс, перейдите в первую консоль и отправьте "зависшему" процессу сигнал на останов (команда **kill-15 PID\_process**).

20. Перейдите в другую консоль и отправьте «непослушному» процессу сигнал **kill-20 PID**. Как реагирует процесс на данный сигнал? Посмотрите в электронном справочнике, что означает данный сигнал.

# УПРАВЛЕНИЕ ПРОЦЕССАМИ В ОС

21. С помощью команды **kill-9 PID** отправьте этому процессу сигнал принудительного завершения. С другой консоли проконтролируйте выполнение команды. Остановился ли процесс? Остался ли он в списке процессов? Какая программа на самом деле перехватывает и исполняет команду **kill-9 PID**?
22. С помощью команды **echo\$PATH** поочередно из консоли администратора и пользователя **user1** выведите список директорий, в которых производится поиск исполняемых файлов, заданных только по имени. В чем заключается различие выведенных списков? Почему в списке **PATH** администратора отсутствует текущий каталог (**.**)? Почему в списке **PATH** пользователя отсутствует каталог **/sbin**? Имеет ли пользователь возможность изменить порядок проверки каталогов для администратора?
23. Попробуйте запустить несколько утилит из второй консоли с правами пользователя (например, **renice-10 PID**, **date-s0**). Как реагирует система на ваши попытки?
24. Перейдите в административную консоль и повторите запуск утилит с правами суперпользователя.

# УПРАВЛЕНИЕ ПРОЦЕССАМИ В ОС

25. Убедитесь в том, что пользователю разрешен запуск указанных утилит. Объясните, почему пользователь не может запустить утилиты с некоторыми «критичными» параметрами? Где, по вашему мнению, расположен механизм контроля заходом запуска (в ядре операционной системы, в командной оболочке, в самой утилите?). Ответ обоснуйте.
26. С правами пользователя скопируйте в свой рабочий каталог один из исполняемых файлов с параметром **SUID** каталога **/bin** (исполняемые файлы выделены цветом и символом \*, а параметр **SUID** отмечен символом «s» в правах владельца на исполнение). Как изменились права доступа к файлу после его копирования?
27. Из второй консоли с правами пользователя скопируйте в свой домашний каталог утилиту, которую разрешено запускать только администратору (например, **chattr**). Копирование производите с параметрами, гарантирующими переход копии во владение пользователю. Убедитесь, что пользователь имеет на скопированный файл все необходимые права. Попробуйте использовать свою копию утилиты по ее назначению (в случае копирования утилиты **chattr** установите дополнительный атрибут **+i** одному из своих файлов). Сделайте выводы.