Лабораторная работа №12

Программирование в командном процессоре ОС UNIX. Командные файлы

Кижваткина Анна Юрьевна

Содержание

# 1 Цель работы

Изучить основы программирования в оболочке ОС UNIX/Linux. Научиться писать небольшие командные файлы.

# 2 Выполнение лабораторной работы

Создаем директорию backup. (рис. 1)

Создание директории

Рис. 1: Создание директории

Создаем файл task1.sh. (рис. 2)

Создание файла

Рис. 2: Создание файла

Пишем скрипт, который при запуске будет делать резервную копию самого себя (то есть файла, в котором содержится его исходный код) в другую директорию backup в вашем домашнем каталоге. При этом файл должен архивироваться одним из архиваторов на выбор zip, bzip2 или tar. Способ использования команд архивации необходимо узнать, изучив справку. (рис. 3)

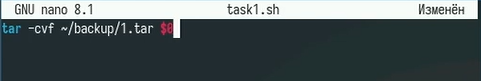


Рис. 3: Программа

Предоставляем полный доступ (чтение, запись и выполнение) к файлу или каталогу для всех пользователей. (рис. 4)

Предоставление доступа

Рис. 4: Предоставление доступа

Проверяем выполнение программы. (рис. 5)

Запуск программы

Рис. 5: Запуск программы

Создаем файл task2.sh. (рис. 6)

Создание файла

Рис. 6: Создание файла

Предоставляем полный доступ (чтение, запись и выполнение) к файлу или каталогу для всех пользователей. (рис. 7)

Предоставление доступа

Рис. 7: Предоставление доступа

Пишем пример командного файла, обрабатывающего любое произвольное число аргументов командной строки, в том числе превышающее десять. Например, скрипт может последовательно распечатывать значения всех переданных аргументов. (рис. 8)

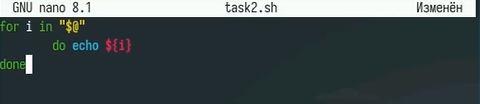


Рис. 8: Программа

Проверяем выполнение программы. (рис. 9)

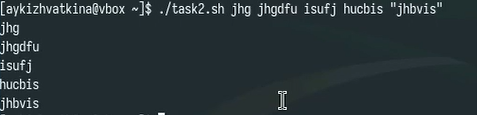


Рис. 9: Запуск программы

Создаем файл task3.sh. (рис. 10)

Создание файла

Рис. 10: Создание файла

Предоставляем полный доступ (чтение, запись и выполнение) к файлу или каталогу для всех пользователей. (рис. 11)

Предоставление доступа

Рис. 11: Предоставление доступа

Пишем командный файл — аналог команды ls (без использования самой этой команды и команды dir). Требуется, чтобы он выдавал информацию о нужном каталоге и выводил информацию о возможностях доступа к файлам этого каталога. (рис. 12)

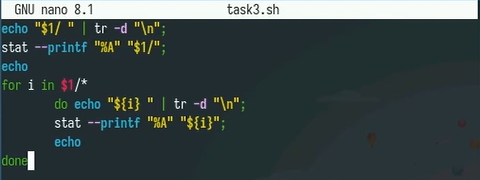


Рис. 12: Программа

Проверяем выполнение программы. (рис. 13)

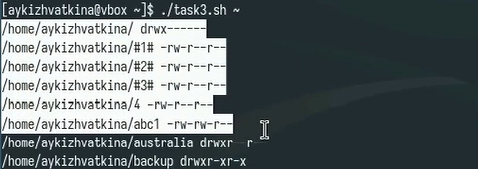


Рис. 13: Запуск программы

Создаем файл task4.sh. (рис. 14)

Создание файла

Рис. 14: Создание файла

Предоставляем полный доступ (чтение, запись и выполнение) к файлу или каталогу для всех пользователей. (рис. 15)

Предоставление доступа

Рис. 15: Предоставление доступа

Пишем командный файл, который получает в качестве аргумента командной строки формат файла (.txt, .doc, .jpg, .pdf и т.д.) и вычисляет количество таких файлов в указанной директории. Путь к директории также передаётся в виде аргумента командной строки. (рис. 16)

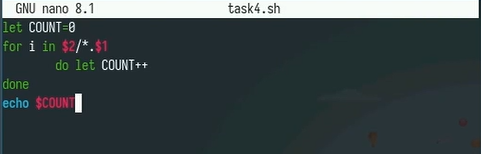


Рис. 16: Программа

Проверяем выполнение программы. (рис. 17)

Запуск программы

Рис. 17: Запуск программы

# 3 Выводы

В ходе данной лабораторной работы мы изучили основы программирования в оболочке ОС UNIX/Linux. Научились писать небольшие командные файлы.