

Лабораторная работа №7

Основы разработки командных файлов в *Linux*

ЦЕЛЬ РАБОТЫ: Изучение команд *Unix* – подобных операционных систем.

Исследование отличий командных файлов в ОС *Linux* от командных файлов ОС семейства *Windows*.

Создайте текстовый файл, и все исследования по работе заносите в него.

Для *Linux* проще всего переключиться в текстовый режим комбинацией клавиш **Alt-Ctrl-F1**. **Alt-Ctrl-F7** - обратное переключение в графический режим. Фактически, комбинации от Alt-Ctrl-F1 до Alt-Ctrl-F6 загружают 6 отдельных приглашений на вход в систему в текстовом режиме, тогда как Alt-Ctrl-F7 - в графическом режиме.

Большинство программ при своём выполнении выводят на экран много текста. Вы можете сохранить этот текст в файл для возможного дальнейшего использования. Чтобы это сделать, воспользуйтесь оператором перенаправления ">".

```
sandbox@laptop:~ > cd /usr
```

```
sandbox@laptop:/usr > ls -l > ~/usr-listing.txt
```

```
sandbox@laptop:/usr > cd
```

```
sandbox@laptop:~ > ls
```

Для того чтобы текстовый файл можно было использовать как командный, существует несколько возможностей. Пусть с помощью редактора создан файл с именем **f1**, содержащий одну строку следующего вида:

Date; pwd; ls

Можно вызвать оболочку **shell** как команду, обозначаемую **sh**, и передать ей файл **f1** и, как аргумент или как перенаправленный вход.

\$sh f1 или

\$sh < f1

В результате выполнения любой из этих команд будет выдана дата, затем имя текущего каталога, а потом содержимое каталога.

Файл можно выполнять и в текущем экземпляре **shell**. Для этого существует специфическая команда «.» (точка).

.f1

Более интересный и удобный вариант работы с командным файлом - это превращение его в выполняемый файл, т.е. просто сделать его командой (запускающей этот файл по его названию), что достигается изменением кода защиты. Для этого надо разрешить выполнение этого файла. Например:

Chmod 711 f1

сделает у этого файла код защиты «RWX X X». Тогда простой вызов **f1** приведет к выполнению тех же трех команд, которые были записаны в файл. Результат будет тот же, если файл с содержимым **Date; pwd; ls** будет представлен в виде:

Date

pwd

ls

так как переход на другую строку также является разделителем в последовательности команд.

Таким образом, выполняемыми файлами могут быть не только файлы, полученные в

результате компиляции и сборки, но и файлы, написанные на языке **shell**. Их выполнение происходит в режиме интерпретации с помощью **shell**-интерпретатора. Начинать командный файл надо всегда с пустой строки или пустого оператора «:».

Работу всех команд операционной системы упрощенно можно представить следующим образом: команда - это некоторое устройство, которое выполняет преобразование входного потока в выходной.

Входной поток в команде может быть задан *четырьмя* способами:

- указанием адреса входного потока справа с команды;
- отсутствием в команде прямого указания на адрес входного потока;
- перенаправлением входного потока с помощью символа «<»;
- передачей во входной поток результата работы предыдущей команды (конвейер, или серфинг или piping)

Выходной поток может быть задан *пятью* способами:

- 1) по умолчанию, без явного указания на выходной поток;
- 2) явным указанием выходного потока;
- 3) перенаправлением выходного потока одной команды на вход другой (символ |);
- 4) перенаправлением в начало файла (символ «>»);
- 5) перенаправлением в конец файла (символ «>>»).

Организация конвейера затрагивает как входной поток, так и выходной.

Чтобы вывести на экран и рассмотреть каталог, в котором находится несколько сотен файлов, необходимо организовать вывод с остановками.

Для вывода всех файлов на экран служит команда **ls -l**. Для ограничения выходного потока этой команды, вывода; каталога отдельными фрагментами, служит команда **more**. Соединив эти две команды в конвейер:

ls -l | more

получим, что содержание команды готовится командой **ls -l**, а выходной поток, который знаком конвейера | направляется на вход команды **more**, выводится на экран с временными паузами.

Другие команды при соединении в конвейер позволяют получать результат, который каждая команда в отдельности получить не позволит.

Задание1 Составьте командный файл, определяющий:

- а) кому принадлежит текущий каталог, дату его создания и состав содержащихся в нём файлов;
- б) общее количество файлов в текущем каталоге; последние 10 раз, когда заходил в систему пользователь с текущим login – ном.

Задание2 Проверьте и опишите, что делают следующие конвейеры команд, (замените, где это нужно команду lp, ввиду недоступности устройства печати):

1. `sort fruits | sp`
2. `sort textfile | lp`
3. `who | grep tty15`
4. `who | sort`
- 5.

6. `ls -l | awk '{print $5}' | sort -n | tail -l`
7. `ls | pr -t -15 -4`
8. `cat *.1994 | grep profits | lp`
9. `sort data_15 | pr | lp`

Задание 3. Опишите разницу в написании, внешнем изображении и выполнении

командных файлов в **Linux** и в Windows XP.