

Лабораторная работа №3

«ПЕРЕНАПРАВЛЕНИЕ ПОТОКОВ ДАННЫХ»

Потоки и файлы

Логически все файлы в системе Linux организованы в непрерывный поток байтов. Любой файл можно свободно копировать и добавлять к другому файлу, так как все файлы организованы одинаково.

Эта логическая организация файлов распространяется на операции ввода и вывода. Данные в операциях ввода и вывода организованы аналогично файлам. Данные, вводимые с клавиатуры, направляются в поток данных, организованный как непрерывная совокупность байтов. Данные, выводимые из команды или программы, также направляются в поток, организованный как непрерывная совокупность байтов. Входной поток данных в ОС Linux называется стандартным вводом, а выходной поток данных - стандартным выводом.

Поскольку стандартный ввод и стандартный вывод организованы так же, как файл, они свободно могут взаимодействовать с файлами. В ОС Linux широко используется переадресация, которая позволяет перемещать данные в файлы и из файлов.

Переадресация стандартного вывода: > и >>

Когда выполняется команда ОС Linux, дающая какую-либо выходную информацию, эта информация направляется в поток данных стандартного вывода. По умолчанию в качестве пункта назначения данных стандартного вывода используется какое-либо устройство, в данном случае экран. Устройства, такие как клавиатура и экран, тоже рассматриваются как файлы. Они принимают и отправляют потоки байтов, имеющих такую же организацию, как и файлы байтового потока. Экран - это устройство, на котором отображается непрерывный поток байтов. По умолчанию стандартный вывод посылает свои данные на экран, где они отображаются.

Для направления стандартного вывода в файл, а не на экран, необходимо использовать оператор переадресации вывода: >. С помощью операции переадресации создается новый файл-адресат. Если он уже существует, то система заменит его содержимое данными стандартного вывода. Для того чтобы этого не произошло, можно установить для shell режим noclobber:

```
set -o noclobber
```

В этом случае операция переадресации существующего файла выполнена не будет. Отменить режим noclobber можно, поставив после оператора переадресации восклицательный знак:

```
cat file1 >! file2
```

Хотя оператор переадресации и имя файла ставятся после команды, перенаправление стандартного потока выполняется не после выполнения команды, а до него. С помощью оператора переадресации создается файл и переадресация организуется до того, как начинают поступать данные со стандартного вывода. Если файл уже существует, он будет разрушен и заменен новым файлом под тем же именем. Команда, генерирующая выходные данные, выполняется только после создания файла переадресации.

Если пользователь попытается использовать одно и то же имя для входного файла команды и переадресованного файла, возникнет ошибка. Так как операция переадресации выполняется первой, входной файл, поскольку он существует, разрушается и заменяется

файлом с тем же именем. Когда команда начинает выполняться, она обнаруживает пустой входной файл. Для добавления стандартного вывода к существующему файлу служит оператор переадресации >>.

Переадресация стандартного ввода: < и <<

Многие команды ОС Linux могут принимать данные со стандартного ввода. Сам стандартный ввод получает данные из устройства или из файла. По умолчанию в качестве устройства для стандартного ввода используется клавиатура. Символы, набираемые на клавиатуре, подаются на стандартный ввод, который затем направляется в команду.

Во многих Linux-системах применяется метод буферизации строк. При буферизации строк, информация посылается на стандартный ввод только после того, как пользователь ввел всю строку.

Стандартный ввод можно переадресовать так же, как и стандартный вывод. Стандартный ввод может приниматься не с клавиатуры, а из файла. Оператор переадресации стандартного ввода: <. Кроме этого для переадресации стандартного ввода применяется механизм "файл здесь":

```
cat << 'слово-признак конца ввода'
> 'текст'
> 'текст'
> 'слово-признак конца ввода'
```

Операции переадресации стандартного ввода и стандартного вывода можно объединять.

Переадресация и пересылка по каналу стандартного потока ошибок: >&, 2>.

При выполнении команд иногда происходят ошибки. Например, пользователь указал неверное количество аргументов или возникла какая-то системная ошибка. Когда возникает ошибка, система выдает специальное сообщение. Как правило, такие сообщения об ошибках отображаются на экране вместе со стандартным выводом. ОС Linux, однако, различает стандартный вывод и сообщения об ошибках. Сообщения об ошибках помещаются еще в один стандартный байтовый поток, который называется стандартным потоком ошибок (диагностики).

Так как сообщения об ошибках направляются в поток, отдельный от стандартного вывода, то в случае переадресации стандартного вывода в файл они все равно появляются на экране.

Стандартный поток ошибок можно переадресовать так же, как и стандартный вывод. Например, сообщения об ошибках можно сохранить в файле для справок. Как и в случае стандартного вывода, пунктом назначения стандартного потока ошибок по умолчанию является экран, однако с помощью специальных операторов переадресации его можно переадресовать в любой файл или устройство.

Для переадресации стандартного потока ошибок в shell предусмотрена специальная возможность. Все стандартные байтовые потоки в операциях переадресации можно обозначать номерами (дескрипторами). Номера 0, 1 и 2 обозначают соответственно стандартный ввод, стандартный вывод и стандартный поток ошибок. Оператор переадресации вывода, >, по умолчанию действует на стандартный вывод, 1. Чтобы переадресовать стандартный поток ошибок, нужно поставить перед оператором переадресации вывода цифру 2.

Стандартный поток ошибок можно дописать в файл, используя цифру 2 и оператор добавления: >>. Для того чтобы переадресовать и стандартный вывод, и стандартный поток ошибок, нужны две операции переадресации и два файла.

В BASH можно ссылаться на стандартный поток по его номеру со знаком "&": &1 обозначает стандартный вывод. Это обозначение можно использовать в операции переадресации для того, чтобы сделать стандартный вывод файлом назначения. Операция переадресации 2>&1 переадресует стандартный поток ошибок на стандартный вывод. В результате стандартный вывод становится файлом назначения для стандартного потока ошибок. Операция переадресации 1>&2 переадресует стандартный ввод в стандартный поток ошибок. По умолчанию входным потоком операции >& является стандартный поток ошибок, а выходным потоком - стандартный вывод. Поэтому, если его использовать в команде, все сообщения о ошибках будут перенаправляться на стандартный вывод.

Программные каналы: |

Иногда возникают ситуации, когда нужно передать данные из одной команды в другую. Другими словами, необходимо послать стандартный вывод одной команды на стандартный ввод другой, а не в файл. Для образования такого соединения в ОС Linux используется так называемый канал. Оператор канала, | (вертикальная черта), помещенный между двумя командами, связывает их стандартные потоки. Стандартный вывод одной команды становится стандартным вводом другой. Выходная информация команды, стоящей перед оператором канала, передается в качестве входной в команду, стоящую за оператором канала.

Программные каналы можно объединять с другими средствами shell, например со специальными символами, проводя таким образом специализированные операции.

Если операция переадресации позволяет просто направлять выходную информацию в файл, то каналы обеспечивают ее пересылку в другую команду Linux. Следует помнить о различии между файлом и командой. Файл - это носитель данных. Вы можете хранить или читать из него данные. Команда - это программа, которая исполняет инструкции. Команда может читать данные из файла и сохранять данные в файле, но во время выполнения ее нельзя рассматривать как файл. По этой причине операция переадресации выполняется с файлами, а не с командами. В процессе переадресации данные посылаются из программы в файл, а не в другую программу. Пунктом назначения операции переадресации могут быть только файлы, но не программы.

Можно, тем не менее, смоделировать процесс конвейерной пересылки с помощью нескольких операций переадресации. Выходная информация одной команды посылается в файл. Команда, записанная в следующей строке, использует этот файл как переадресованный ввод.

Каналы работают со стандартным выводом команды независимо от того, что подается на этот вывод. Пересылаться по каналу из одной команды в другую может содержимое целого файла и даже нескольких файлов.

Стандартным вводом, посылаемым по каналу в команду, можно более эффективно управлять с помощью аргумента стандартного ввода, -. Дефис, используемый в команде в качестве аргумента, обозначает стандартный ввод.

Допустим, нужно напечатать файл с именем его каталога в начале списка. Команда pwd выдает имя каталога, а команда cat выдает содержимое файла. В данном случае команде cat нужно использовать в качестве входной информации и файл, и стандартный ввод, пересланный по каналу из команды pwd. Команда cat будет иметь два аргумента: стандартный ввод, обозначенный дефисом, и имя выводимого на печать файла.

```
pwd | cat - file | lpr
```

Каналы и переадресация: команда tee

Для того, чтобы переадресовать стандартный вывод в файл и одновременно воспроизвести эту информацию на экран необходимо использовать команду tee. Команда tee копирует стандартный вывод в файл. В качестве аргумента она использует имя нового файла, в который копируется стандартный вывод. Это все равно что скопировать

содержимое стандартного вывода и один его экземпляр переадресовать в файл, а другой отправить дальше (часто - на экран).

Переадресацией в сочетании с каналами необходимо пользоваться осторожно. Переадресация стандартного вывода задает файл назначения для него. Стандартный вывод записывается и сохраняется в этом файле. После записи никакой информации для пересылки по каналу в другую команду не остается. Переадресация может производиться в конце последовательности программных каналов, но не внутри этой последовательности.

Задание для выполнения

1. Вывести любое сообщение с помощью команды `echo` перенаправив вывод:

- в несуществующий файл с помощью символа `>`;
- в несуществующий файл с помощью символа `>>`;
- в существующий файл с помощью символа `>`;
- в существующий файл с помощью символа `>>`;

Объяснить результаты.

2. Переадресовать стандартный ввод для команды `cat` на файл.

3. Перенаправить ввод для команды `cat` с помощью механизма "файл здесь", указав слово признака конца ввода:

- стандартным способом;
- не первым в строке;
- не единственным в строке.

Объяснить результаты.

4. Вывести сообщение с помощью команды `echo` в канал ошибок. Создать файл `myscript`:

```
#!/bin/sh
echo stdout
echo stderr>&2
exit 0
```

Запустить его:

- без перенаправления (`sh myscript`);
- перенаправив стандартный вывод в файл, просмотреть содержимое файла (`sh myscript > file1`);
- перенаправить стандартный канал ошибок в существующий и несуществующий файлы с помощью символов `>` и `>>` (а тут и дальше уже сами :));
- перенаправив стандартный вывод в файл 1, стандартный канал ошибок - в файл 2;
- перенаправив стандартный вывод и стандартный канал ошибок в файл 3;
- перенаправив стандартный вывод в файл 4 с помощью символа `>`, а стандартный канал ошибок в файл 4 с помощью символа `>>`;

Объяснить результаты.

5. Вывести третью строку из последних десяти строк отсортированного в обратном порядке файла `/etc/group`.

6. Реализовать конвейер, который выводит десять строк: пять первых строк из файла `/etc/passwd` в неотсортированном виде, после чего пять последних из того же файла в отсортированном виде, используя команду `tee`.

7. Подсчитать при помощи конвейера команд количество блочных устройств ввода-вывода, доступных в системе.

Отчет

Как и в других работах, отчет по проделанной работе представляется преподавателю в стандартной форме: на листах формата А4, с титульным листом (включающим тему, фио, номер зачетки и пр.), целью, ходом работы и выводами по выполненной работе. Каждое задание должно быть отражено в отчете следующим образом: 1) что надо было сделать, 2) как это сделали, 3) что получилось, и, в зависимости от задания – 4) почему получилось именно так, а не иначе. При наличии заданий с вариантами необходимо указать свой вариант, и расчет его номера, а также всё вышеуказанное для данного варианта задания.