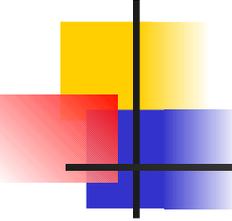


BIF703

Redirection Continued: Pipes



Redirection

Recall from the previous slides we defined `stdin`, `stdout`, and `stderr` and we learned how to redirect these to files.

STDIN - Standard Input (redirection symbol: `<`, `0<`)

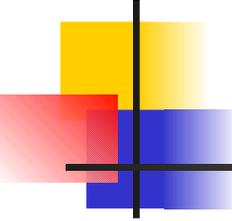
- Data read from a file or terminal (eg. keyboard)

STDOUT - Standard Output

- Data (output) as a result of a command or program executed. (redirection symbols: `>`, `1>`, `>>`, `1>>`)

STDERR - Standard Error

- Error message as a result of improper syntax of command or factors that lead to failure of task. (redirection symbols: `2>`, `2>>`)

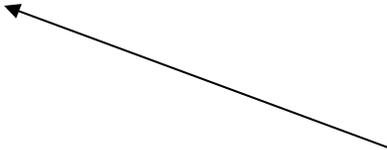


Redirection

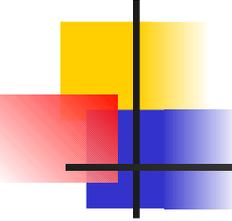
Let's consider using redirection to solve a series of tasks.

Task: Assuming your current directory contains filename only consisting of digits, re-arrange the listing in reverse numerical order. Below is an short listing:

```
[userprompt] ls  
1 101 11 2 20 3
```



Notice that filenames are in alphanumeric order...



Redirection

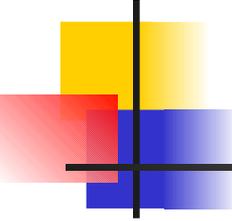
We can solve this problem in the following manner:

1. Issue a Linux command to get a short listing and redirect stdout to a **temporary file**.
2. Issue another Linux command to sort that **temporary file** in reverse numerical order.

```
[userprompt] ls > /tmp/myTempFile  
[userprompt] sort -rn /tmp/myTempFile  
101  
20  
11  
3  
2  
1
```

It is good to store temporary files in the **/tmp** directory to avoid overwriting files...

Notice that filenames are now in reverse numerical order...

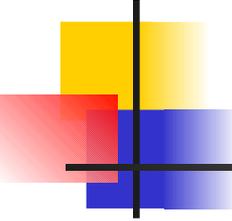


Redirection

There are some inefficiencies with this “solution”:

1. You should remove the temporary file (i.e. clean-up after yourself)!
2. You are required then to issue 3 separate Linux commands – Isn't there an easier way?

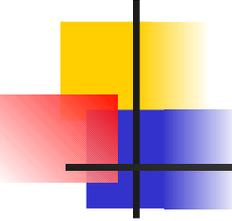
YES – Use Pipes



Pipes (|)



- Pipes are used to send the **stdout** of the first command as **stdin** into the second command.
- The purpose of using pipes is to use commands after the pipe that act as “**filters**” to **modify stdout** until you reach the desired result.
- Pipes have the advantage of modifying standard output to **achieve a task without having to create temporary files**.
- In order to successfully use pipes, **the user must understand what stdout is produced from a command, and which commands accept stdin. and produce stdout.**



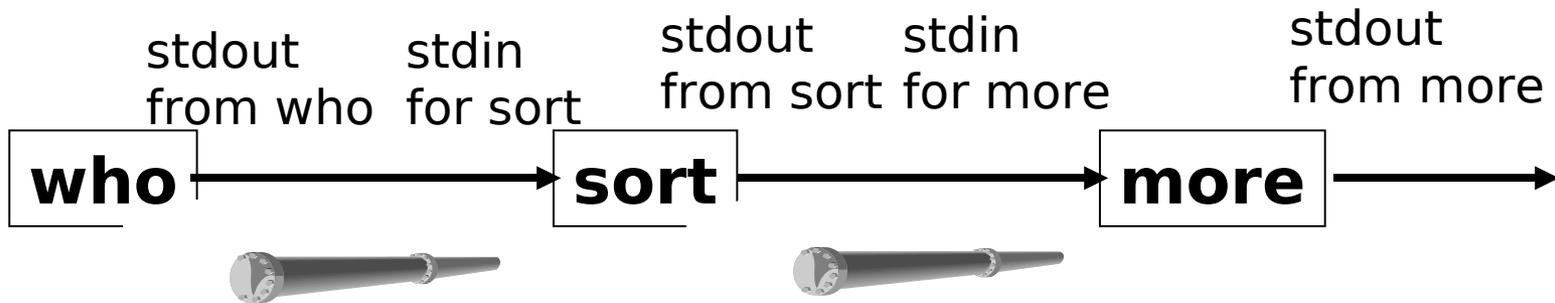
Pipes (|)

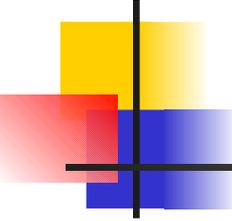
Demonstration of Pipes:

who | sort | more

The pipe symbol | is generated by pressing **SHIFT** \

Result:





Pipes (|)

Let's do the earlier task, but instead of redirecting stdout to a temporary file, let's use a pipe instead:

```
[userprompt] ls | sort -rn
```

```
101
```

```
20
```

```
11
```

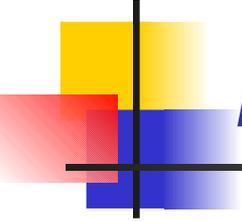
```
3
```

```
2
```

```
1
```

Remember, the stdout from ls becomes the stdin for the sort command (no filename required!)

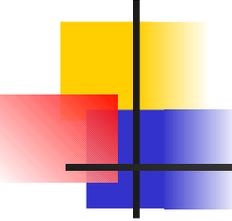
Same result as before, but only 1 command was issued!



Pipes (|)

Sample Word Problem:

- Write a Linux pipe-line command to display a list of users on the current server that only allow message reception. This list should be sorted in reverse alphanumeric order by the first column.



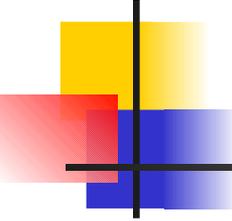
Pipes (|)

Sample Word Problem:

- Write a Linux pipe-line command to display a list of users on the current server that only allow message reception. This list should be sorted in reverse alphanumeric order by the first column.

Solution:

- Step 1: Identify the Linux commands to be used



Pipes (|)

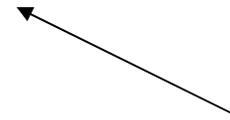
Sample Word Problem:

who -T



- Write a Linux pipe-line command to display a list of users on the current server that only allow message reception. This list should be sorted in reverse alphanumeric order by the first column.

sort -r

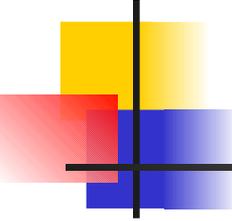


grep "+"



Solution:

- Step 1: Identify the Linux commands to be used:
who -T | grep "+" | sort -r



Pipes (|)

Solution / Continued:

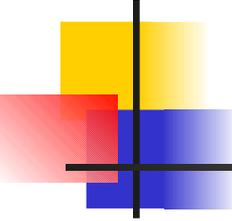
- Step 2: Place the Linux commands in the proper order using the pipe symbol

Takes stdout from
who -T command and
only displays lines with +

who -T | grep "+" | sort -r

↑
Gives the most
stdout with column of
+ allows message reception
- denies message reception

↑
Takes stdout from
grep "+" modification and
sorts in reverse order



Pipes (|)

Additional Practice:

- Write a Linux pipeline command to display records in the `/etc/passwd` file, sorted alphabetically by the first field, and have the output display the first screenfull.
- Write a Linux pipeline command to display only lines 3 to 7 of a ten line file called `data.txt`
- Write a Linux pipeline command to display a listing of files in your current directory, sorted in reverse alphanumeric order, display on the screen and save the same output to a file called `listing`. (Hint: use the online manual on the `tee` command)



Additional Resources

- This slide-show provides the minimum amount of redirection using pipes for this course.
- For a more comprehensive discussion of miscellaneous Linux commands, please refer to the [PDF](#) file for [Week9_Lesson2](#).
- Here are some Related-Links for Interest Only:

Redirection (Includes using pipes):

- <http://www.pclab.uni-bk.ac.yu/linux/ch02/019-022.html>