

1/4

UNIT 5

Standard I/O and Pipes



Rev RH033-RHEL4-1

Copyright © 2005 Red Hat, Inc.

For use only by a student enrolled in a Red Hat training course taught by Red Hat, Inc. or a Red Hat Certified Training Partner. Any other use is a violation of U.S. and international copyrights. No part of this publication may be photocopied, duplicated, stored in a retrieval system, or otherwise reproduced without prior written consent of Red Hat, Inc. If you believe Red Hat training materials are being improperly used, copied, or distributed please email training@redhat.com or phone toll-free (USA) +1 866 826 2994 or +1 (919) 754 3700.

UNIT 5: Objectives

- Understand standard I/O channels
- Understand file redirection
- Understand pipes



2

Rev RH033-RHEL4-1

Copyright © 2005 Red Hat, Inc.

For use only by a student enrolled in a Red Hat training course taught by Red Hat, Inc. or a Red Hat Certified Training Partner. Any other use is a violation of U.S. and international copyrights. No part of this publication may be photocopied, duplicated, stored in a retrieval system, or otherwise reproduced without prior written consent of Red Hat, Inc. If you believe Red Hat training materials are being improperly used, copied, or distributed please email training@redhat.com or phone toll-free (USA) +1 866 626 2994 or +1 (919) 754 3700.

UNIT 5: Agenda

- Standard I/O channels
- File redirection
- Pipes



Rev RH033-RHEL4-1

Copyright © 2005 Red Hat, Inc.

For use only by a student enrolled in a Red Hat training course taught by Red Hat, Inc. or a Red Hat Certified Training Partner. Any other use is a violation of U.S. and international copyrights. No part of this publication may be photocopied, duplicated, stored in a retrieval system, or otherwise reproduced without prior written consent of Red Hat, Inc. If you believe Red Hat training materials are being improperly used, copied, or distributed please email training@redhat.com or phone toll-free (USA) +1 866 626 2994 or +1 (919) 754 3700.

Standard Input and Output

- Linux provides three I/O channels to processes
 - Standard input - keyboard is default
 - Standard output - terminal window is default
 - Standard error - terminal window is default



Rev RH033-RHEL4-1

Copyright © 2005 Red Hat, Inc.

For use only by a student enrolled in a Red Hat training course taught by Red Hat, Inc. or a Red Hat Certified Training Partner. Any other use is a violation of U.S. and international copyrights. No part of this publication may be photocopied, duplicated, stored in a retrieval system, or otherwise reproduced without prior written consent of Red Hat, Inc. If you believe Red Hat training materials are being improperly used, copied, or distributed please email training@redhat.com or phone toll-free (USA) +1 866 626 2994 or +1 (919) 754 3700.

One of the most important features of Linux (and UNIX) is the streaming nature of data known as standard input, standard output, and standard error. In general, this allows the input from a program to come from any source, and the output to go to any source. In addition, the output from one command can be fed directly to the input of another command.

Standard output, by default, is the screen or terminal window. Most commands send their output to standard output without explicitly being told to do so. When standard output is sent to a destination other than the screen, such as a file, one is *redirecting standard output*.

Standard input defaults to the keyboard. Most commands use files as their source of input, but will accept standard input from other sources, via redirection.

A third data stream is standard error. This is a secondary output stream that carries warnings, usage messages, error messages and other "out-of-band" information, in an output stream distinct from standard output. This error stream is normally sent to the screen, but may be redirected elsewhere.

These streams are abbreviated as `stdin` (called file descriptor number 0), `stdout` (file descriptor number 1), and `stderr` (file descriptor number 2). The fact that there are two output channels allows separation of error messages from normal output. For example, error messages could be saved in a file with the normal output going to the monitor.

Redirecting Input and Output

- Standard Input, Output, and Error can be reconnected to alternate locations
 - Shell redirection operators allow standard I/O channels to be redirected to/from a file
 - Pipes allow standard I/O channels to be connected to the input or output of *programs*



5

Rev RH033-RHEL4-1

Copyright © 2005 Red Hat, Inc.

For use only by a student enrolled in a Red Hat training course taught by Red Hat, Inc. or a Red Hat Certified Training Partner. Any other use is a violation of U.S. and international copyrights. No part of this publication may be photocopied, duplicated, stored in a retrieval system, or otherwise reproduced without prior written consent of Red Hat, Inc. If you believe Red Hat training materials are being improperly used, copied, or distributed please email training@redhat.com or phone toll-free (USA) +1 866 625 2994 or +1 (919) 754 3700.

The standard output of commands, which ordinarily displays on the terminal, can be redirected into a file or piped into another command.

Standard error, which also ordinarily displays on the terminal, can be redirected into a file. Although it is also possible to pipe standard error into a file using some fairly complex syntax, this is generally not done.

Standard input, ordinarily coming from the keyboard, can be redirected from a file. More commonly, the standard output of one command can be piped into the standard input of another command.

Common Redirection Operators

>	command > file	Directs standard output of <i>command</i> to <i>file</i>
>>	command >> file	Appends standard output of <i>command</i> to <i>file</i>
<	command < file	<i>command</i> receives its input from <i>file</i>
2>	command 2> file	Error messages from <i>command</i> are directed to <i>file</i>
2>>	command 2>> file	Error messages from <i>command</i> are appended to <i>file</i>

Piping

	command1 command2	Pipes the standard output of <i>command1</i> into the standard input of <i>command2</i>
--	---------------------	---

Redirecting Output

- In order to study redirecting standard output and error, we will use the `find` command.

```
find /etc -name passwd
```
- This command will search for all files named `passwd` in `/etc` and its subdirectories.
- By default both the standard output and standard error are displayed on the screen.



6

Rev RH033-RHEL4-1

Copyright © 2005 Red Hat, Inc.

For use only by a student enrolled in a Red Hat training course taught by Red Hat, Inc. or a Red Hat Certified Training Partner. Any other use is a violation of U.S. and international copyrights. No part of this publication may be photocopied, duplicated, stored in a retrieval system, or otherwise reproduced without prior written consent of Red Hat, Inc. If you believe Red Hat training materials are being improperly used, copied, or distributed please email training@redhat.com or phone toll-free (USA) +1 866 626 2994 or +1 (919) 754 3700.

Example output

```
[bob@station2 tmp]$ find /etc -name passwd
/etc/passwd
find: /etc/default: Permission denied
/etc/pam.d/passwd
```

The specific error messages you receive may vary, depending on the particular packages you have installed.

Redirecting Standard Output

- Redirect standard output with `>`
 - Example: redirect standard output to a file:
`find /etc -name passwd > findresult`
- Standard error is still displayed on the screen



7

Rev RH033-RHEL4-1

Copyright © 2005 Red Hat, Inc.

For use only by a student enrolled in a Red Hat training course taught by Red Hat, Inc. or a Red Hat Certified Training Partner. Any other use is a violation of U.S. and international copyrights. No part of this publication may be photocopied, duplicated, stored in a retrieval system, or otherwise reproduced without prior written consent of Red Hat, Inc. If you believe Red Hat training materials are being improperly used, copied, or distributed please email training@redhat.com or phone toll-free (USA) +1 866 826 2994 or +1 (919) 754 3700.

Example output with stdout redirected to a file and stderr displayed to the terminal:

```
[bob@station2 tmp]$ find /etc -name passwd > findresult
find: /etc/default: Permission denied
```

```
[bob@station2 tmp]$ cat findresult
/etc/passwd
/etc/pam.d/passwd
```

Overwriting or Appending

- If the target file of file redirection with `>` already exists, the existing file will be overwritten
- To append data to an existing file, use `>>` to redirect instead of `>`



8

Rev RH033-RHEL4-1

Copyright © 2005 Red Hat, Inc.

For use only by a student enrolled in a Red Hat training course taught by Red Hat, Inc. or a Red Hat Certified Training Partner. Any other use is a violation of U.S. and international copyrights. No part of this publication may be photocopied, duplicated, stored in a retrieval system, or otherwise reproduced without prior written consent of Red Hat, Inc. If you believe Red Hat training materials are being improperly used, copied, or distributed please email training@redhat.com or phone toll-free (USA) +1 866 626 2994 or +1 (919) 754 3700.

Redirect output and create/overwrite the output file

```
[bob@station2 tmp]$ find /etc -name passwd > output
find: /etc/default: Permission denied
```

Redirect output and append to the output file

```
[bob@station2 tmp]$ find /etc -name passwd >> output
find: /etc/default: Permission denied
find: /etc/cups/certs: Permission denied
```

View the output file

```
[bob@station2 tmp]$ cat output
/etc/passwd
/etc/pam.d/passwd
/etc/passwd
/etc/pam.d/passwd
```

Redirect output and overwrite the output file

```
[bob@station2 tmp]$ find /etc -name passwd > output
find: /etc/default: Permission denied
```

View the output file

```
[bob@station2 tmp]$ cat output
/etc/passwd
/etc/pam.d/passwd
```


Redirecting Standard Error

- Redirect standard error with 2>
- Example: redirect standard error to a file:

```
find /etc -name passwd 2> finderrors
```

- Standard output is displayed on the screen
- Redirect further standard error, appending to the same file, with 2>>



9

Rev RH033-RHEL4-1

Copyright © 2005 Red Hat, Inc.

For use only by a student enrolled in a Red Hat training course taught by Red Hat, Inc. or a Red Hat Certified Training Partner. Any other use is a violation of U.S. and international copyrights. No part of this publication may be photocopied, duplicated, stored in a retrieval system, or otherwise reproduced without prior written consent of Red Hat, Inc. If you believe Red Hat training materials are being improperly used, copied, or distributed please email training@redhat.com or phone toll-free (USA) +1 866 629 2994 or +1 (919) 754 3700.

Redirect stderr to the finderrors file

```
[bob@station2 tmp]$ find /etc -name passwd 2> finderrors
/etc/passwd
/etc/pam.d/passwd
```

View the finderrors file

```
[bob@station2 tmp]$ cat finderrors
find: /etc/default: Permission denied
find: /etc/cups/certs: Permission denied
```

Redirect some more stderr to the finderrors file

```
[bob@station2 tmp]$ find /tmp -name passwd 2>> finderrors
```

View the finderrors file

```
[bob@station2 tmp]$ cat finderrors
find: /etc/default: Permission denied
find: /tmp/orbit-root: Permission denied
```

Redirecting Both Standard Output and Error

- Redirection of standard output and standard error can be performed simultaneously:
`find / -name passwd 2> errs > results`
- Each I/O channel can be redirected to different files, or to the same file:
`find / -name passwd > alloutput 2>&1`
(or) `find / -name passwd &> alloutput`



Rev RH033-RHEL4-1

Copyright © 2005 Red Hat, Inc.

For use only by a student enrolled in a Red Hat training course taught by Red Hat, Inc. or a Red Hat Certified Training Partner. Any other use is a violation of U.S. and international copyrights. No part of this publication may be photocopied, duplicated, stored in a retrieval system, or otherwise reproduced without prior written consent of Red Hat, Inc. If you believe Red Hat training materials are being improperly used, copied, or distributed please email training@redhat.com or phone toll-free (USA) +1 866 626 2994 or +1 (919) 754 3700.

Redirect stderr to the errs file and redirect stdout to the results file

```
[bob@station2 tmp]$ find /etc -name passwd 2> errs > results
```

View the errs file

```
[bob@station2 tmp]$ cat errs
find: /etc/default: Permission denied
```

View the results file

```
[bob@station2 tmp]$ cat results
/etc/passwd
/etc/pam.d/passwd
```

Redirect both stderr and stdout to the alloutput file

```
[bob@station2 tmp]$ find /etc -name passwd > alloutput 2>&1
```

View the alloutput file

```
[bob@station2 tmp]$ cat alloutput
/etc/passwd
find: /etc/default: Permission denied
/etc/pam.d/passwd
```

Redirecting Input

- Redirect standard input with `<`
- Some commands only operate on standard input
 - `tr` doesn't accept filenames as arguments - it requires its input to be redirected from somewhere
 - `tr 'A-Z' 'a-z' < .bash_profile`
 - This command will translate the uppercase characters in `.bash_profile` to lowercase



Rev RH033-RHEL4-1 Copyright © 2005 Red Hat, Inc.
For use only by a student enrolled in a Red Hat training course taught by Red Hat, Inc. or a Red Hat Certified Training Partner. Any other use is a violation of U.S. and international copyrights. No part of this publication may be photocopied, duplicated, stored in a retrieval system, or otherwise reproduced without prior written consent of Red Hat, Inc. If you believe Red Hat training materials are being improperly used, copied, or distributed please email training@redhat.com or phone toll-free (USA) +1 866 826 2994 or +1 (919) 754 3700.

Many Linux commands, like `cat`, will take their input from a file if one is given as an argument. Commands such as these usually accept either a filename or standard input. The following two commands produce the same output:

```
$ cat filename.txt
$ cat < filename.txt
```

A few commands, like the character translator `tr`, do not accept filenames as arguments and must have their input redirected.

Example output

```
[bob@station2 tmp]$ cat textfile.txt
TEXT TYPED IN ALL CAPITALS IS OFTEN DIFFICULT TO READ.
TRANSLATING SUCH TEXT INTO LOWER CASE CAN IMPROVE
ITS LEGIBILITY.

[bob@station2 tmp]$ tr 'A-Z' 'a-z' < textfile.txt
text typed in all capitals is often difficult to read.
translating such text into lower case can improve
its legibility.
```


Using Pipes To Connect Processes

- Pipes (the | character) let you redirect output from one command to become the input to another command

```
$ ls /usr/lib | less
```

- Can create *pipelines* - a powerful feature of Linux

```
$ cut -f1 -d: passwd | sort -r | less
```



Rev R-033-RHEL4-1

Copyright © 2005 Red Hat, Inc.

For use only by a student enrolled in a Red Hat training course taught by Red Hat, Inc. or a Red Hat Certified Training Partner. Any other use is a violation of U.S. and international copyrights. No part of this publication may be photocopied, duplicated, stored in a retrieval system, or otherwise reproduced without prior written consent of Red Hat, Inc. If you believe Red Hat training materials are being improperly used, copied, or distributed please email training@redhat.com or phone toll-free (U.S.A.) +1 866 626 2994 or +1 (919) 754 3700.

Two of the basic tenets of UNIX philosophy are: make small programs that do one thing well, and expect the output of every program to become the input to another, as yet unknown, program. The use of pipes let you leverage the effects of these two design principles to create extremely powerful chains of commands to achieve your desired result. Most command-line programs that operate on files can also accept input on standard in, and output to standard out. Some commands like `tr` are designed to only operate within a pipe (can not operate on files directly).

Any command that writes to standard output can be used on the left-hand side of a pipe.

Any command that reads from standard input can be used on the right-hand side of a pipe.

Multiple commands can be chained together with pipes.

Example output

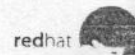
```
[bob@station2 man]$ pwd
/usr/share/man
```

```
[bob@station2 man]$ ls -C | tr 'a-z' 'A-Z'
MAN1  MAN3  MAN5  MAN7  MAN9  PT_BR  WHATIS
MAN2  MAN4  MAN6  MAN8  MANN  TMAC.H
```

In the example above, lower case letters of the alphabet from the file listing are converted to upper case letters.

Useful Pipe Targets

- **less** displays input one page at a time
`ls -l | less`
- **mail** sends input via email:
`ls -l | mail -s "Files" bob@example.com`
- **lpr** sends input to the printer
`ls -l | lpr`
- **xargs** converts input to argument list
`cat files_to_delete.txt | xargs rm -f`



13

Rev RH033-RHEL4-1

Copyright © 2005 Red Hat, Inc.

For use only by a student enrolled in a Red Hat training course taught by Red Hat, Inc. or a Red Hat Certified Training Partner. Any other use is a violation of U.S. and international copyrights. No part of this publication may be photocopied, duplicated, stored in a retrieval system, or otherwise reproduced without prior written consent of Red Hat, Inc. If you believe Red Hat training materials are being improperly used, copied, or distributed please email training@redhat.com or phone toll-free (USA) +1 866 626 2994 or +1 (919) 754 3700.

Earlier we demonstrated using the `less` command to display a text file one screenfull at a time. But what if the output of a command takes up more than a screen? You can also use `less` in these situations to buffer and view the command's output by piping the command into it.

The traditional `mail` command is rather old fashioned for a mailer. But, it is extremely useful for mailing the standard output of a command. The `-s` option allows you to specify a subject for the email.

You can also pipe output to the `lpr` command. This will send the output to your system's default printer. If you want to use a non-default printer you can specify one with the `-P` option:

```
ls -l | lpr -P some_printer
```

All of the above examples demonstrate sending the output of one command to the standard input of another command. But what if you wanted to convert the output of one command into part of the command line for another? For example, in the example shown in this slide we have a file called `files_to_delete.txt` that contains a list of filenames. Suppose it contained the following files:

`/home/student/letter.txt`

`/home/student/images/pic.jpg`

`/tmp/foo`

The command shown in the slide would combine the lines of this file into a space-delimited list of arguments to the `rm -f` command. In other words, the command would be equivalent to running:

```
rm -f /home/student/letter.txt /home/student/images/pic.jpg /tmp/foo
```

.tee

- Lets you *tee* a pipe: redirect output to a file while still piping it to another program
`$ set | tee set.out | less`
- In example, output from `set` is written to file `set.out` while also being piped to `less`



Rev RH033-RHEL4-1

Copyright © 2005 Red Hat, Inc.

For use only by a student enrolled in a Red Hat training course taught by Red Hat, Inc. or a Red Hat Certified Training Partner. Any other use is a violation of U.S. and international copyrights. No part of this publication may be photocopied, duplicated, stored in a retrieval system, or otherwise reproduced without prior written consent of Red Hat, Inc. If you believe Red Hat training materials are being improperly used, copied, or distributed please email training@redhat.com or phone toll-free (USA) +1 866 628 2994 or +1 (919) 754 3700.

Uses of tee

`tee` is useful to save the output at various stages from a long sequence of pipes. For example:

```
$ command | tee stage1.out | sort | tee stage2.out | uniq -c | \
tee stage3.out | sort -r | tee stage4.out | less
```


End of UNIT 5

- Questions and answers
- Summary
 - Standard I/O
 - File Redirection
 - Pipes



Rev RH033-RHEL4-1 Copyright © 2005 Red Hat, Inc. For use only by a student enrolled in a Red Hat training course taught by Red Hat, Inc. or a Red Hat Certified Training Partner. Any other use is a violation of U.S. and international copyrights. No part of this publication may be photocopied, duplicated, stored in a retrieval system, or otherwise reproduced without prior written consent of Red Hat, Inc. If you believe Red Hat training materials are being improperly used, copied, or distributed please email training@redhat.com or phone toll-free (USA) +1 866 626 2994 or +1 (919) 754 3700.

Important files covered in this Unit:

Important commands covered in this Unit:

tee