

Viewing and Editing Files

Here are some simple commands that display the contents of files to the screen.

`cat file` - Display the entire contents of file.

`more file` - Browse through a text file. Press the `Spacebar` to advance to the next page. Press `Enter` to advance to the next line. Type `q` to quit viewing the file. Commands are based on the `vi` editor, which is covered in the next section.

`less file` - Like `more` but allows backward movement and pattern searches.

`head file` - Output the beginning (or top) portion of file.

`tail file` - Output the ending (or bottom) portion of file.

This is how you might examine a file named `file.txt` with the commands `cat`, `tail`, and `more`.

```
$ cat file.txt
This is the first line.
This is the second.
Here is some more interesting text.
Knock knock.
Who's there?
More filler text.
The quick brown fox jumps over the lazy dog.
The dog was rather impressed.
Roses are red,
Violets are blue,
All my base are belong to you.
Finally, the 12th and last line.
$ head file.txt
This is the first line.
This is the second.
Here is some more interesting text.
Knock knock.
Who's there?
More filler text.
The quick brown fox jumps over the lazy dog.
The dog was rather impressed.
Roses are red,
Violets are blue,
$ tail file.txt
Here is some more interesting text.
Knock knock.
Who's there?
More filler text.
The quick brown fox jumps over the lazy dog.
The dog was rather impressed.
Roses are red,
Violets are blue,
All my base are belong to you.
Finally, the 12th and last line.
$ more file.txt
Here is some more interesting text.
Knock knock.
Who's there?
More filler text.
...
$
```

By default `head` and `tail` only display ten lines. You can override this behavior and tell them to display a specified number of lines. The format is `-n` where `n` is the number of lines you want to display. If you only want to display the first line of a file use `head -1 file`. Want to display the last three lines? Then run `tail -3 file`.

```
$ head -2 file.txt
This is the first line.
This is the second.
$ tail -1 file.txt
Finally, the 12th and last line.
$
```

Viewing Files In Real Time

Using `cat` can be a fine way to view files that have fairly static content. However, if you are trying to keep up with changes that are being made in real time to a file, `cat` is not the best choice. A good example of files that can change often and rapidly are log files. For example, you may need to start a program and look at that program's log file to see what it is doing. For this case, use the `tail -f file` command.

`tail -f file` - Follow the file. Displays data as it is being written to the file.


```
$ tail -f /opt/app/var/log.txt
Oct 10 16:41:17 app: [ID 107833 user.info] Processing request 7680687
Oct 10 16:42:28 app: [ID 107833 user.err] User pat denied access to admin
functions
...
```

Editing Files

Nano

If you need to edit a file right now and do not want to spend any time learning obscure editor commands, use `nano`. `Nano` is a clone of `pico`, so if for some reason the `nano` command is not available, `pico` probably is. It's not as powerful as some other editors, but it's definitely easier to learn.

When you start `nano` you will see the file's contents and a list of commands at the bottom of the screen. To run the commands, replace the caret symbol (^) with the `ctrl` key. For example, to exit `nano` type `ctrl-x`.



```
GNU nano 2.0.6           File: file.txt

This is the first line.
This is the second.
Here is some more interesting text.
Knock knock.
Who's there?
More filler text.
The quick brown fox jumps over the lazy dog.
The dog was rather impressed.
Roses are red,
Violets are blue,
All my base are belong to you.
Finally, the 12th and last line.

[ Read 12 lines ]

^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text ^T To Spell
```

Editing in `nano` is quite easy. The up and down arrow keys will take you to the previous or next lines as expected. The right and left arrow keys let you navigate forwards and backwards on the same line. Simply type the desired text into the editor. To save the file, type `ctrl-o`. If you forget to save the file before you exit, `nano` will ask you if you want to save the file. To learn more type `ctrl-g` for help.

<http://www.LinuxTrainingAcademy.com>