

# Editing, vi and Configuration Files

## Unix/IP Preparation Course

July 19, 2009  
Eugene, Oregon, USA  
*hervey@nsrc.org*



# Goals

- Be able to edit a file using vi
- Use some of vi's more advanced features
- Begin to understand the “language” of configuration files
- Use alternate editors: ee, joe, pico, emacs, xemacs, gedit, etc.

# Some of our philosophy about vi

- It's available!
- Wait, what was that? Oh yeah, it's available!
- It's has some very powerful features.
- It's ubiquitous in Unix (`visudo`, `vipw`, `vigr`, etc.)
- Not that hard to learn after initial learning curve.
- Impress your friends and family with your arcane knowledge of computers.

# Why is vi “so hard to use”?

Like all things it's not really – once you are used to how it works.

The ***critical*** vi concept:

1. vi has two modes
2. These modes are ***insert*** and ***command***

Let's see how we use these...

# vi command and insert modes

## Swapping modes

- When you open a file in vi you are in *command mode* by default.
- If you wish to edit the file *you need to switch to insert mode first*.
- To exit *insert mode* press the ESCape key.
- If you get used to this concept you are halfway done to becoming a competent vi user.

# vi insert mode

Two common ways to enter insert mode upon opening a file include:

- Press the “i” key to start entering text directly after your cursor.
- Press the “o” key to add a new line *below* you cursor and to start adding text on the new line.
- Remember, to exit *insert mode* press the ESCape key at any time.

# vi command mode

There are many, many commands in vi, but some of the most common and useful are:

- Press “**x**” to delete a character at a time.
- Press “**dd**” quickly to press the line you are on.
- Press “/”, and text to search for and press <ENTER>.
  - Press “n” to find the next occurrence of text.
  - Press “N” to find previous occurrences of text.

# Saving a file or “How to exit vi”

1. In vi press the *ESC*ape key to verify you are in command mode.
2. Depending on what you want to do press:
  - **:w** → write the file to disk
  - **:wq** → write the file to disk, then quit
  - **:q** → quit the file (only works if no changes)
  - **:q!** → quit and lose any changes made
  - **:w!** → override r/o file permission if you are owner or root and write the file to disk.
  - **:w!q** → override r/o file permission if you are owner or root and write the file to disk and quit.



# Speed-Up your Config File Editing!

1. In vi press the *ESC*ape key to verify you are in command mode.
2. To search for the first occurrence of something:
  - `/string` → press <ENTER>
  - “n” → press “n” for each following occurrence
  - “N” → press “N” for each previous occurrence
3. To replace *all* occurrences of a string in a file:
  - `:%s/old_string/new_string/g`
4. To replace *all* occurrences of a string in a file:
  - `:%s/old_string/new_string/gc`

# Speed things up some more!

1. In vi press the *ESCape* key to verify you are in command mode.
2. Go directly to a specific line number
  - `:NN` → press <ENTER>. If NN=100, go to line 100
3. Go to start/end of a line
  - press *Home* or press *End* on your keyboard
4. Go to top/bottom of a file:
  - press *ctrl-Home* or press *ctrl-End* on your keyboard
5. Undo the last change you made (in command mode)
  - press “u”

# Editing configuration files

There are patterns to how configuration files work:

- The most common comment character is “#”.
- After that you'll see “/\* .... \*/” or “//”.
- There are a few others, but they are less common.

# Editing configuration files cont.

Some configuration files have lots of comments and few directives. Others are the opposite.

Blocks of configuration may be indicated in a programmatic manner, i.e.:

```
<VirtualHost *>  
    <SubSection>  
        directive  
        directive  
    </SubSection>  
</VirtualHost>
```

# Editing configuration files cont.

Another standard is to do the following:

```
## comment
```

```
## comment
```

```
# default setting=off
```

To change the default do:

```
default setting=on
```

# Editing configuration files cont.

## Things to watch out for:

- Spaces
- Quotes and single quotes: “directive” or 'directive'
- Caps or CamelCase syntax
  - Localhost=”myhost”
  - LocalHost=”myhost”
- Line end indicator (: or ;)
- New-line or continuation character “\”.

# Conclusion

vi's most confusing feature is that it works in two modes and you must switch between them.

## Questions?