



Les secrets de vim

>>> Liste des commandes de vim

Description :

Ce cours a pour but de vous apprendre à utiliser le logiciel Vim. Je détaillerai la liste des commandes et des raccourcis possible sous Vim. Vim peut paraître compliqué mais une fois maîtrisé, on ne s'en passe plus.

Les secrets de vim

>>> Liste des commandes de vim

Sommaire :

- I) Première fois
 - II) Présentation
 - 1) Exemple
 - III) Les commandes de base
 - 1) Modifications
 - 2) Gestion des fichiers
 - 3) Déplacements, recherches et remplacements
 - IV) Comparaison de fichiers
 - V) Options avancées
 - 1) Couleur dans Vim
 - 2) Activer la souris
-

Vim est l'un des principaux éditeurs de texte en mode console sur Debian, c'est aussi l'un des plus performant et plus puissant. Depuis la dernière version de Debian (Squeeze) Vim n'est plus installé par défaut lors de l'installation du système. Pas de panique, une simple commande va vous permettre de l'installer :

```
aptitude install vim
```

Si vous pensiez que Vim ne pouvez pas comme n'importe quel éditeur de texte faire du copier/coller, annuler la dernière action, rechercher, remplacer, etc. Vous vous trompez !

I) Première fois

Comme je l'ai dit ci-dessus, Vim est un logiciel qui peut paraître compliqué. Votre première fois avec Vim va sûrement vous paraître un enfer.

Ne surtout pas se décourager !!!

Pour utiliser Vim, il vous faudra passer un peu de temps. Votre formation se fera en l'utilisant.

II) Présentation

Vim fonctionne suivant trois modes de saisie :

- Un mode **saisie** : permettant d'éditer normalement un fichier.
- Un mode **commande** : permettant la saisie de commandes Vim ou de commandes Shell.
- Un mode **visuel** : permettant de sélectionner des parties de texte.

Au lancement de Vim, celui -ci se trouve par défaut en mode **commande**. La saisie de texte ne vous ait

donc pas possible.

Pour entrer en mode **saisie**, appuyer sur la touche **"i"** de votre clavier. Dès lors que vous avez appuyé sur la touche, vous passez en mode saisie et Vim vous affiche en bas de l'écran — **INSERTION** —. Vous pouvez vous déplacer avec les flèches dans votre fichier et modifier ou ajouter du texte.

Pour quitter le mode **saisie** et revenir au mode **commande**, appuyer sur **ESC** de votre clavier.

Pour entrer en mode **visuel**, appuyer sur la touche **"v"** de votre clavier, Vim affiche en bas de votre écran — **VISUEL** —.

Pour quitter le mode **saisie** et revenir au mode **commande**, appuyer sur **ESC** de votre clavier.

1) Exemple :

- Taper la commande : **vim**

Le logiciel Vim se lance.

- Appuyer sur la touche **"i"**

Vous passez en mode saisie.

- Taper une phrase : **Petite phrase pour Vim**

- Appuyer sur la touche **"ESC"**

Pour quitter le mode saisie.

- Appuyer sur la touche **"v"**

Vous passez en mode visuel.

- Appuyer sur la touche **"ESC"**

Pour quitter le mode visuel.

- Taper : **:q !**

Vous quittez Vim sans sauvegarder.

III) Les commandes de base

Voici les principales commandes pour utiliser Vim.

1) Modifications

Les commandes pour modifier le fichier :

- Couper du texte :

Commandes	Descriptions
dd	Coupe la ligne sur laquelle se trouve le curseur.
ndd	Coupe n ligne à partir de l'emplacement du curseur, si on veut couper 4, on tape 4dd .

Commandes	Descriptions
d0	Coupe tous les caractères avant le curseur de la ligne où vous vous trouvez.
D ou d\$	Coupe tous les caractères après le curseur de la ligne où vous vous trouvez.

- Copier du texte :

Commandes	Descriptions
yy	Copie la ligne sur laquelle se trouve le curseur.
nyy	Copie n ligne à partir de l'emplacement du curseur, si on veut couper 4, on tape 4yy .
y0	Copie tous les caractères avant le curseur de la ligne où vous vous trouvez.
y\$	Copie tous les caractères après le curseur de la ligne où vous vous trouvez.

- Coller du texte :

Commandes	Descriptions
p	Colle les lignes copiées ou coupées.

- Autres commandes utiles :

Commandes	Descriptions
a	Passage en mode saisie et positionne le curseur après le caractère sur lequel se trouve le curseur.
o	Passage en mode saisie et ajoute une ligne vide sous la ligne où se trouve votre curseur.
u	Annule la dernière modification, équivalent de CTRL+Z.
:undolist	Affiche la liste des dernières modifications.
:undo n	Permet de choisir de sauter directement à une modification donnée. Le terme n est le numéro de modifications.
:earlier 15s	Récupère la version du fichier telle qu'elle était il y a 15 secondes.
:later 15s	Même chose mais dans l'autre sens

2) Gestion des fichiers

Les commandes suivantes gèrent le fichier :

- Sauvegarde du fichier :

Commandes	Descriptions
ZZ	Sauvegarde le fichier courant et quitte Vim.
:x	Sauvegarde le fichier courant et quitte Vim.
:wq	Sauvegarde le fichier courant et quitte Vim.
:w nom_de_fichier	Sauvegarde le fichier en utilisant le nom de fichier entré en paramètre.

- Quitter Vim :

Commandes	Descriptions
:q	Quitte l'éditeur.
:q !	Quitte l'éditeur sans sauvegarder

- Autres commandes utiles :

Commandes	Descriptions
:e nom_de_fichier	Ouvre le fichier dont le nom est passé en paramètre.
: !commande shell	Permet de lancer une commande du shell sans sortir de l'éditeur. Par exemple : !ifconfig permet d'afficher les paramètres réseaux, appuyer sur Entrée pour revenir à Vim

3) Déplacements, recherches et remplacements

Vous voulez déplacer votre curseur plus rapidement, pas de problème voici quelques commandes :

- Se déplacer :

Commandes	Descriptions
gg	Déplace le curseur sur la première ligne du fichier.
G	Déplace le curseur sur la dernière ligne du fichier.
nG	Déplace le curseur sur la énième ligne du fichier.
zz	Place la ligne courante au centre de la fenêtre.

- Rechercher :

Commandes	Descriptions
/terme	Recherche le terme dans le texte situé après le curseur.
?terme	Recherche le terme dans le texte situé avant le curseur.
n	Répète la dernière recherche dans le texte situé après le curseur.
N	Répète la dernière recherche dans le texte situé avant le curseur.
#	Recherche le mot sur lequel pointe le curseur, dans le texte situé après.
*	Recherche le mot sur lequel pointe le curseur, dans le texte situé avant.

- Remplacements

Commandes	Descriptions
plage s/find/remplace/[options]	Remplace l'expression find par remplace sur la plage indiquée. Une plage est constituée d'un numéro de ligne de départ, d'une virgule et d'un numéro de ligne de fin. Pour spécifier l'ensemble des lignes du fichier, vous pourrez utiliser le raccourci % . Les commandes optionnelles sont g , qui permet d'appliquer le remplacement à toutes les occurrences d'une même ligne sinon seulement la première occurrence est remplacée. Et l'option c qui demande une confirmation des remplacements.

IV) Comparaison de fichiers

Vous voulez comparer deux fichiers de configuration pour rechercher les différences pas de problème, avec Vim tout est possible.

On commence par ouvrir un des deux fichiers :

```
vim <fichier1>
```

Ensuite taper la commande suivante :

```
:vert diffsplit <fichier2>
```

Une séparation verticale de la fenêtre va apparaître permettant de voir les différences. On peut transmettre la modification d'un fichier à l'autre via la commande :

```
:diffput
```

- Permet de transmettre la modification.

Ou

```
:diffget
```

- Permettant de récupérer la modification.

Pour naviguer entre les modifications, on utilise les commandes :

```
|c
```

- Pour aller à la modification suivante.

Ou

```
|C
```

- Pour aller à la modification précédente.

V) Options avancées

1) Couleur dans Vim

Première étape :

Commencez par éditer le fichier `/etc/vim/vimrc` :

```
vim /etc/vim/vimrc
```

Deuxième étape :

On dé-commente la ligne `syntax on` et la ligne `set background=dark` :

On doit obtenir ceci :

```
" Vim5 and later versions support syntax highlighting. Uncommenting the next
" line enables syntax highlighting by default.
syntax on

" If using a dark background within the editing area and syntax highlighting
" turn on this option as well
set background=dark
```

2) Activer la souris

Pour activer la souris rien de plus simple :

- On édite le fichier **.vimrc** :

```
vim /etc/vim/vimrc
```

et on ajoute la ligne suivante :

```
set mouse=a
```

25 septembre 2011 -- N.Salmon -- article_224.pdf



Idum