

Nano est un éditeur de texte, pas un traitement de texte !

En sciences, le terme « nano » représente une toute petite unité. Par exemple, un atome a une taille d'environ 0,1 nanomètre.

Si l'éditeur de texte que je vais vous présenter s'appelle Nano, c'est parce qu'il est tout petit. Il s'agit d'un programme très simple comparé à Vim et Emacs et il nous conviendra tout à fait pour démarrer. Il possède assez peu de fonctions par rapport aux deux autres logiciels (qui peuvent devenir de véritables machines de guerre) mais suffisamment pour commencer à vous débrouiller avec un éditeur de texte.

Savez-vous vraiment ce qu'est un éditeur de texte ? Ne le confondez-vous pas avec un traitement de texte ?

Un **éditeur de texte** est un programme qui permet de modifier des fichiers de texte brut, sans mise en forme (gras, italique, souligné...). Sous Windows, on dispose d'un éditeur de texte très basique : le Bloc-Notes. Sous Linux, on a le choix entre Nano, Vim, Emacs et bien d'autres, sachant qu'au moins un de ceux-là est installé par défaut sur la plupart des distributions.

Un **traitement de texte** est fait pour rédiger des documents mis en forme. Sous Windows, Word est le plus célèbre traitement de texte ; sous Linux, on possède l'équivalent : Open Office Writer. Ces programmes ne peuvent être utilisés qu'en mode graphique, la console ne permettant pas vraiment de faire de la mise en forme.

Quand a-t-on besoin d'un éditeur de texte ?

Chaque fois que vous devez éditer un fichier de texte brut. Sous Windows, vous avez l'habitude de voir des fichiers de texte brut au format `.txt`. Sous Linux, vous savez que l'extension importe peu (on peut trouver des fichiers en texte brut sans extension).

Les éditeurs de texte sont parfaits pour les programmeurs en particulier : ils permettent d'éditer des fichiers `.c`, `.cpp`, `.h`, `.rb`, `.py`, etc. (En fonction de votre langage de programmation.)

Même si vous ne programmez pas, vous aurez besoin d'utiliser un éditeur de texte pour modifier des fichiers de configuration. Ces fichiers n'ont pas d'extension particulière, mais à force vous apprendrez à les reconnaître.

Après avoir appris à utiliser Nano, nous nous ferons les dents sur nos premiers fichiers de configuration : le `nanorc` et le `bashrc`. Ce sera l'occasion pour vous de

personnaliser votre Nano et votre console.:-)

Découverte de Nano

Le nom complet de Nano est « GNU nano », en référence au projet GNU dont je vous ai parlé dans le tout premier chapitre. Il s'agit d'un logiciel qui s'inspire de « pico », un éditeur de texte plus ancien qui se voulait lui aussi très simple d'utilisation.

Pour démarrer le logiciel, il vous suffit simplement de taper nano dans la console :

```
nano
```

L'éditeur Nano s'ouvre immédiatement (figure suivante).



Nano

Dès lors, vous pouvez commencer à taper du texte (exemple sur la figure suivante).

```
GNU nano 2.0.6          Nouvel espace          Modifié
Salut les Zéros, ça va ?
Je teste nano ! Un deux... un deux...
^G Aide      ^O Écrire   ^R Lire fich.^Y Page préc.^K Couper    ^C Pos. cur.
^X Quitter   ^J Justifier ^W Chercher  ^V Page suiv.^U Coller   ^T Orthograp.
```

Nano : écriture de texte

C'est aussi simple que cela !

Ne riez pas, je précise qu'il « suffit de taper du texte » car ce n'est pas aussi simple sous d'autres éditeurs, comme Vim par exemple.

Les raccourcis clavier de Nano

En bas de votre écran, vous pouvez voir un espace d'aide (figure suivante). Que signifie-t-il exactement ?

Il s'agit d'un aide-mémoire pour vous rappeler à tout moment les commandes principales que vous pouvez lancer sous Nano.

```
^G Aide      ^O Ecrire   ^R Lire fich.^Y Page préc.^K Couper    ^C Pos. cur.
^X Quitter   ^J Justifier ^W Chercher  ^V Page suiv.^U Coller   ^T Orthograp.
```

Aide de Nano

Le symbole ^ signifie Ctrl (la touche Contrôle de votre clavier). Ainsi, pour quitter Nano, il suffit de taper Ctrl + X.

Voici les raccourcis les plus importants :

- Ctrl + G : afficher l'aide ;
- Ctrl + K : couper la ligne de texte (et la mettre dans le presse-papier) ;

- `Ctrl + U` : coller la ligne de texte que vous venez de couper ;
- `Ctrl + C` : afficher à quel endroit du fichier votre curseur est positionné (numéro de ligne...);
- `Ctrl + W` : rechercher dans le fichier ;
- `Ctrl + O` : enregistrer le fichier (écrire) ;
- `Ctrl + X` : quitter Nano.

Vous pouvez vous déplacer dans le fichier avec les flèches du clavier ainsi qu'avec les touches `Page Up` et `Page Down` pour avancer de page en page (les raccourcis `Ctrl + Y` et `Ctrl + V` fonctionnent aussi).

Si l'aide-mémoire vous encombre, vous pouvez gagner de la place en appuyant sur `Échap` puis sur `X`. Vous pouvez l'afficher de nouveau avec la même suite de touches.

La recherche

La combinaison de touches `Ctrl + W` lance une recherche dans le fichier (figure suivante).

```

GNU nano 2.0.6          Nouvel espace          Modifié
Salut les Zéros, ça va ?
Je teste nano ! Un deux... un deux...

Recherche:
^G Aide      ^Y Prem. lig.^R Remplacer ^W Début paraM-C Resp.cassM-R Exp. ratio.
^C Annuler   ^V Dern. Lig.^T Aller lig.^O Fin para. M-B ->Arrière^P Précédente

```

Recherche dans Nano

Il vous suffit d'écrire le mot que vous recherchez (figure suivante)...

```
GNU nano 2.0.6          Nouvel espace          Modifié
Salut les Zéros, ça va ?
Je teste nano ! Un deux... un deux...

Recherche: deux
^G Aide      ^Y Prem. lig.^R Remplacer ^W Début paraM-C Resp.cassM-R Exp. ratio.
^C Annuler   ^V Dern. Lig.^T Aller lig.^O Fin para. M-B ->Arrière^P Précédente
```

Recherche dans Nano

... puis de taper Entrée (figure suivante).

```
GNU nano 2.0.6          Nouvel espace          Modifié
Salut les Zéros, ça va ?
Je teste nano ! Un deux... un deux...

[ La recherche a fait le tour ]
^G Aide      ^O Écrire    ^R Lire fich.^Y Page préc.^K Couper    ^C Pos. cur.
^X Quitter   ^J Justifier^W Chercher  ^V Page suiv.^U Coller    ^T Orthograp.
```

Recherche dans Nano

Le curseur est automatiquement positionné à la première occurrence trouvée. Si le

curseur est à la fin, la recherche recommence du début.

Si vous voulez sortir du mode recherche, tapez `Ctrl + C` (Annuler).

Si vous souhaitez aller au résultat suivant (au « deux » suivant), faites à nouveau `Ctrl + W` pour lancer une recherche. La recherche précédente est sauvegardée et apparaît entre crochets. Si vous voulez rechercher le même mot (et donc aller au résultat suivant), tapez juste `Entrée` sans écrire de mot à rechercher (figure suivante).

```
GNU nano 2.0.6          Nouvel espace          Modifié
Salut les Zéros, ça va ?
Je teste nano ! Un deux... un deux...

Recherche [deux]:
^G Aide      ^Y Prem. lig.^R Remplacer ^W Début paraM-C Resp.cassM-R Exp. ratio.
^C Annuler   ^V Dern. Lig.^T Aller lig.^O Fin para. M-B ->Arrière^P Précédente
```

Recherche dans Nano

Enregistrer et quitter

Pour enregistrer à tout moment, faites `Ctrl + O`.

Si vous essayez de quitter (`Ctrl + X`) sans enregistrer auparavant, un message vous demandera si vous voulez sauvegarder (figure suivante).

```
GNU nano 2.0.6          Nouvel espace          Modifié
Salut les Zéros, ça va ?
Je teste nano ! Un deux... un deux...

Sauver l'espace modifié (RÉPONDRE « Non » EFFACERA LES CHANGEMENTS) ?
O Oui
N Non          ^C Annuler
```

Confirmation de sortie de Nano

Si vous appuyez sur la touche o, vous passerez en mode enregistrement.

Si vous appuyez sur la touche n, Nano quittera sans enregistrer.

Si vous utilisez la combinaison `Ctrl + C`, vous annulerez votre demande de sortie de Nano et ne quitterez donc pas le logiciel.

En appuyant sur o, vous vous retrouvez en mode enregistrement. Tapez juste le nom du fichier que vous voulez créer puis pressez Entrée (figure suivante).

```
GNU nano 2.0.6          Nouvel espace          Modifié
Salut les Zéros, ça va ?
Je teste nano ! Un deux... un deux...

Nom du fichier à écrire: salut.txt
^G Aide          ^T Parcourir    M-M Format Mac   M-P Ajout (au début)
^C Annuler       M-D Format DOS  M-A Ajout (à la fin) M-B Copie de sécu.
```

Enregistrement dans Nano

Après ça, Nano sera fermé et vous retrouverez votre bonne vieille ligne de commandes.

Les paramètres de la commande Nano

Lorsque vous appelez Nano dans la ligne de commandes, vous pouvez spécifier plusieurs paramètres. Le plus courant est d'indiquer en paramètre le nom du fichier qu'on veut ouvrir. Ainsi :

```
nano salut.txt
```

... ouvrira le fichier `salut.txt` que l'on vient de créer.

Si le fichier n'existe pas, il sera automatiquement créé par Nano lors du premier enregistrement.

À part ça, la commande `nano` accepte de nombreux paramètres. Pour vous, j'en ai sélectionné trois qui me semblent faire partie des plus utiles.

- **-m** : autorise l'utilisation de la souris sous Nano. En console, oui, oui. Vous pouvez vous en servir pour cliquer avec votre souris sur la zone de texte où vous voulez placer votre curseur.
- **-i** : indentation automatique. L'alinéa (tabulations) de la ligne précédente sera

respecté lorsque vous irez à la ligne. Très utile lorsque vous éditez un fichier de code source.

- **-A** : active le retour intelligent au début de la ligne. Normalement, lorsque vous appuyez sur la touche **Origine** (aussi connue sous le nom de **Home**) située à côté de la touche **Fin**, le curseur se repositionne au tout début de la ligne. Avec cette commande, il se positionnera après les alinéas. Comme **-i**, il s'agit d'une option utile avant tout pour les programmeurs.

Si je veux lancer Nano avec toutes ces options à la fois, je peux donc écrire :

```
nano -miA salut.txt
```

Pourquoi `.nanorc` ?

Vous savez maintenant utiliser Nano. Comme vous avez pu le voir, ce n'est pas très compliqué. Il suffit d'apprendre un peu les raccourcis clavier les plus utiles et on peut rapidement s'en servir.

Justement... et si on utilisait Nano pour quelque chose d'utile ? Non parce que bon, le fichier `salut.txt` est sympa, mais ça ne va pas nous faire avancer.

Alors pour l'occasion, je me suis dit que j'allais vous faire éditer quelques fichiers de configuration. Par exemple, il existe un fichier de configuration de Nano qui indique toutes vos préférences. Celui-ci s'appelle `.nanorc`.

La plupart des fichiers de configuration commencent par un point. Cela permet de « cacher » le fichier quand on fait un `ls`. Bien entendu, comme vous devriez maintenant le savoir, les fichiers cachés peuvent toujours être affichés en utilisant le paramètre `-a` : `ls -a`.

Chaque utilisateur de la machine peut créer son propre fichier de configuration `.nanorc` dans son répertoire personnel (home). Chez moi, ce fichier doit être placé à la position : `/home/mateo21/.nanorc`. Ce fichier est lu par Nano à chaque fois que vous le démarrez.

Je viens de regarder la liste des fichiers de mon home, mais même en incluant les fichiers cachés avec `-a` je ne vois pas de fichier appelé `.nanorc` !

En effet, il se peut que le fichier `.nanorc` n'existe pas chez vous. Si tel est le cas, Nano sera chargé avec les options par défaut.

Création du `.nanorc`

Pas de `.nanorc` ? Pas de problème, il suffit de le créer. On peut par exemple faire ceci :

```
nano .nanorc
```

Cette commande ouvre Nano. Comme le fichier `.nanorc` n'existe pas, un document vide est ouvert (figure suivante). Le fichier `.nanorc` sera créé lorsque vous enregistrerez.

```
GNU nano 2.0.6      Fichier : .nanorc
[ Nouveau fichier ]
^G Aide      ^O Écrire    ^R Lire fich.^Y Page préc.^K Couper      ^C Pos. cur.
^X Quitter   ^J Justifier ^W Chercher  ^V Page suiv.^U Coller     ^T Orthograp.
```

Dans ce fichier, vous devez écrire une commande par ligne.

Chaque commande commence par un **set** (pour activer) ou un **unset** (pour désactiver) suivi de l'option qui vous intéresse.

Par exemple, pour activer la souris, écrivez :

```
set mouse
```

Ainsi Nano sera automatiquement chargé avec la prise en charge de la souris. Vous n'aurez pas besoin de réécrire systématiquement le paramètre `-m` qu'on a vu tout à l'heure.

On peut faire de même pour éviter d'avoir à taper à chaque fois les paramètres `-i` et `-A` avec d'autres séries de `set`. Au final, on écrira ceci :

```
set mouse
set autoindent
set smarthome
```

Enregistrez le fichier avec `Ctrl + O`. Comme vous avez déjà mentionné le nom du fichier en paramètre lors de l'ouverture de Nano, celui-ci sera automatiquement écrit pour vous (figure suivante).

```
GNU nano 2.0.6      Fichier : .nanorc      Modifié
set mouse
set autoindent
set smarthome

Nom du fichier à écrire: .nanorc
^G Aide      ^T Parcourir      M-M Format Mac      M-P Ajout (au début)
^C Annuler    M-D Format DOS    M-A Ajout (à la fin) M-B Copie de sécu.
```

Vous pouvez ensuite faire `Ctrl + X` pour quitter Nano.

Je vous rappelle que pour que ces options soient prises en compte, il faut démarrer une nouvelle session de Nano (c'est pour ça que la souris n'a pas automatiquement fonctionné dès que vous avez enregistré le fichier).

Si vous relancez Nano ensuite, vous pouvez constater que la souris fonctionne et que les options d'indentation automatique et de retour à la ligne intelligent sont elles aussi opérationnelles. :-)

Le `nanorc` global et la coloration syntaxique

Ce fichier `.nanorc` dans votre home est très pratique car il vous permet de définir vos propres options. Mais si vous avez dix utilisateurs sur votre machine et que vous voulez activer le support de la souris pour tout le monde, vous n'allez quand même pas créer un fichier `.nanorc` pour chacun !

Il existe un fichier `nanorc` « global » qui est pris en compte pour tout le monde. Celui-ci est situé dans `/etc/nanorc` (attention : il n'y a pas de point devant, cette fois.)

Ce fichier ne peut être modifié que par root. Je vous conseille donc de l'ouvrir avec un `sudo` (ou dans une console en root si vous avez fait `sudo su` avant) :

```
sudo nano /etc/nanorc
```

Normalement, ce fichier existe déjà. Comme vous pouvez le constater sur la figure suivante, il est bien rempli.

```
GNU nano 2.0.6      Fichier : /etc/nanorc
## Sample initialization file for GNU nano.
##
## Please note that you must have configured nano with --enable-nanorc
## for this file to be read! Also note that this file should not be in
## DOS or Mac format, and that characters specially interpreted by the
## shell should not be escaped here.
##
## To make sure a value is disabled, use "unset <option>".
##
## For the options that take parameters, the default value is given.
## Other options are unset by default.
##
## Quotes inside string parameters don't have to be escaped with
## backslashes. The last double quote in the string will be treated as
## its end. For example, for the "brackets" option, "'>]]'" will match
## ", ', ), >, ], and }.
##
## Use auto-indentation.
# set autoindent

Lecture de 263 lignes
^G Aide      ^O Écrire    ^R Lire fich.^Y Page préc.^K Couper     ^C Pos. cur.
^X Quitter   ^J Justifier ^W Chercher  ^V Page suiv.^U Coller    ^T Orthograp.
```

Il sert en fait de fichier d'exemple. Toutes les options disponibles dans un `.nanorc` sont présentes, mais elles sont précédées d'un `#` qui signifie qu'il s'agit d'un commentaire. Les commentaires sont ignorés par Nano.

Le début du fichier vous explique (en anglais) que c'est un fichier d'initialisation d'exemple de Nano.

Après le petit blabla d'introduction, vous avez la liste des options disponibles. Toutes sont commentées.

La première est `autoindent`.

```
# set autoindent
```

Supprimez juste le `#` pour décommenter la ligne et donc pour activer l'indentation automatique pour tous les utilisateurs.

```
set autoindent
```

Vous pouvez parcourir le fichier à la recherche d'options intéressantes que vous voulez activer.

Vers la fin, vous verrez une section appelée « `color setup` », qui commence par ces

lignes-là :

```
## Nanorc files
# include "/usr/share/nano/nanorc.nanorc"

## C/C++
# include "/usr/share/nano/c.nanorc"

## HTML
# include "/usr/share/nano/html.nanorc"
```

Je vous invite à décommenter toutes les lignes d'`include`. Cela permettra d'activer la coloration « intelligente » de vos fichiers selon leur type. Vous pourrez ainsi avoir des fichiers HTML colorés, des fichiers C colorés, des fichiers nanorc colorés, etc.

Enregistrez le fichier puis quittez Nano.

Si vous avez une erreur lors de l'enregistrement, cela signifie que vous n'avez pas ouvert le fichier en root. Seul root a le droit de modifier ce fichier. Fermez Nano et relancez-le avec un `sudo` cette fois.