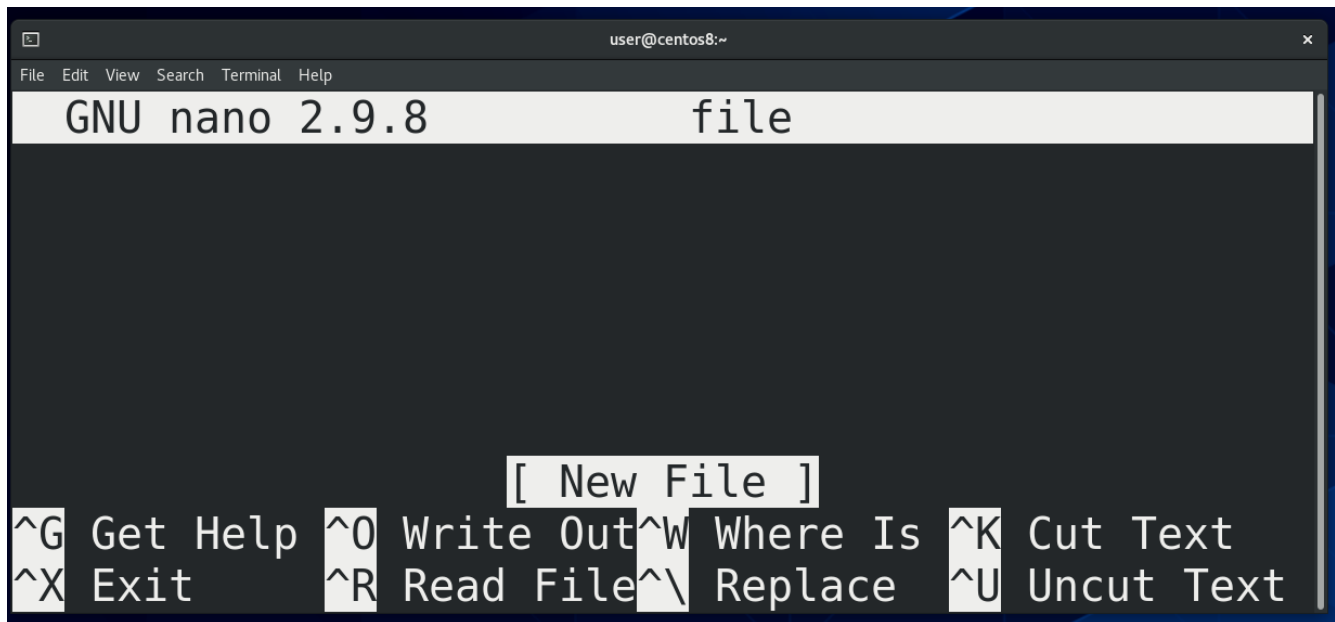


Для редактирования текстовых файлов вам нужен текстовый редактор. Самые известные – vi (и его современная реализация vim), emacs и nano. Объяснением одного nano не обойтись – хоть он и попроще, но очень важно уметь работать с vi и от этого никуда не деться. nano не всегда предустановлен, а без умения работы с vi может сложиться ситуация, что вы и nano не сможете установить. Нет, теоретически, можно и без vi обойтись, но это потребует у вас много времени – легче научиться хотя бы базово работать с этой программой. И не то чтобы vi хуже или сильно сложнее – большинство админов и программистов как раз таки предпочитают vim. Я рассмотрю обе программы, чем пользоваться – решайте сами. Но уметь работать с vi нужно в любом случае.



```
user@centos8:~
File Edit View Search Terminal Help
GNU nano 2.9.8 file

[ New File ]

^G Get Help ^O Write Out ^W Where Is ^K Cut Text
^X Exit ^R Read File ^_ Replace ^U Uncut Text
```

Начнём с nano. Программа простая – вы пишете nano и имя файла, который вы хотите создать или изменить – nano file. Открывается программка и здесь вы можете вводить текст, изменять его. Для перемещения используются стрелки. Чтобы быстро перемещаться между словами можно зажать Ctrl и использовать стрелки, чтобы перейти в начало строки – Ctrl+A, чтобы в конец – Ctrl+E. Внизу есть подсказки по горячим клавишам. Значок рядом с буквами(^ - карет) обозначает Ctrl, например, Ctrl+G для небольшого гайда.

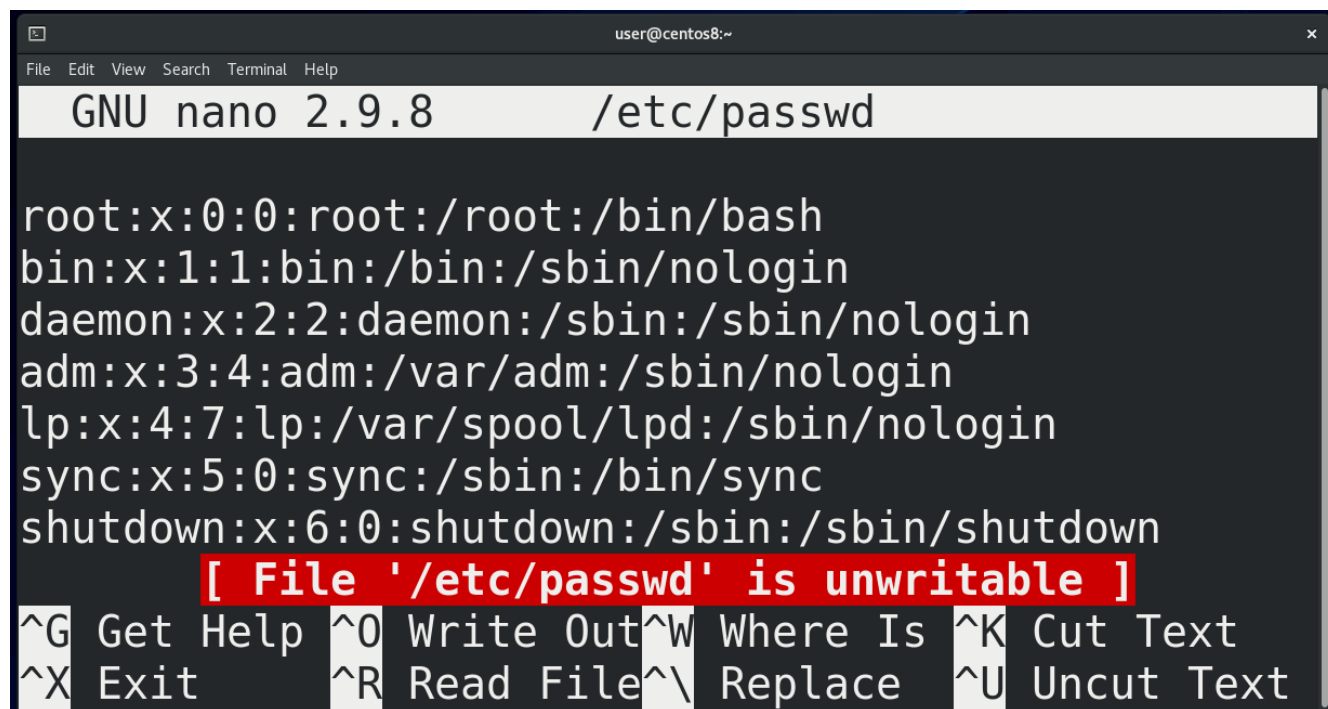
```
user@centos8:~  
File Edit View Search Terminal Help  
Main nano help text  
  
M-U          Undo the last operation  
M-E          Redo the last undone operation  
  
M-A  (^6)    Mark text starting from the  
          cursor position  
M-6  (M-^)   Copy current line (or marked  
          region) and store it in cutbuffer  
  
^L Refresh  ^W Where Is  ^P Prev Line ^Y Prev Page  
^X Close    M-W WhereIs  ^N Next Line ^V Next Page
```

В гайде кроме знака Ctrl также встречается M – это не буква М, а клавиша мета. Скорее всего, у вас на клавиатуре её нет, поэтому её заменяет либо Alt, либо клавиша Win. Допустим, комбинация Alt-U. Как видите, внизу есть подсказка – Ctrl+X - закрыть это окно.

```
user@centos8:~  
File Edit View Search Terminal Help  
GNU nano 2.9.8          file          Modified  
  
A  
b  
d  
c  
e  
  
File Name to Write: file  
  
^G Get Help  M-D DOS Form  M-A Append  M-B Backup Fil  
^C Cancel    M-M Mac Form   M-P Prepend ^T To Files
```

Разбирать все горячие клавиши я не буду, но давайте пройдемся по основным. Начнём с сохранения – написали какой-то текст, хотим сохранить. Нажимаем Ctrl+O – внизу появляется поле, где можно указать новое имя для файла, либо оставить то имя, которое мы указывали, когда запускали nano. Нас это имя устраивает, поэтому нажимаем Enter и видим, что внизу

появилась надпись, где сказано, сколько линий у нас в файле. Чтобы выйти – Ctrl+X. Если перед закрытием мы сделали какие-то изменения, то у нас появится вопрос – сохранить изменения или нет – тут пишем Y или N, или Ctrl+C, всё как указано на подсказке.



```
GNU nano 2.9.8 /etc/passwd

root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown

[ File '/etc/passwd' is unwritable ]

^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text
^X Exit      ^R Read File ^\ Replace   ^U Uncut Text
```

Давайте обратимся к файлу /etc/passwd - nano /etc/passwd. Как видите, внизу надпись нас предупреждает, что этот файл невозможно редактировать – потому что у нас нет на это прав. Поэтому просто скопируем этот файл к себе в домашнюю директорию - cp /etc/passwd ~. С копией файла у нас не будет никаких проблем, так как эта копия принадлежит нам. Откроем копию файла - nano ~/passwd и убедимся, что теперь этой ошибки нет.

```
user@centos8:~  
File Edit View Search Terminal Help  
GNU nano 2.9.8 /home/user/passwd Modified  
9 mail:x:8:12:mail:/var/spool/mail:/sbin/nologin  
10 operator:x:11:0:operator:/root:/sbin/nologin  
11 games:x:12:100:games:/usr/games:/sbin/nologin  
12 ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin  
13 nobody:x:65534:65534:Kernel Overflow User:/:/s$  
14 dbus:x:81:81:System message bus:/:/sbin/nologin  
15 systemd-coredump:x:999:997:systemd Core Dumper$  
line 12/71 (16%), col 1/44 (2%), char 431/3776 (11  
^G Get Help ^O Write Out ^W Where Is ^K Cut Text  
^X Exit ^R Read File ^\ Replace ^U Uncut Text
```

Нередко нам бывает нужно сориентироваться, в какой строчке мы сейчас находимся. Для этого нужно нажать Ctrl+C – появится подсказка – на какой линии вы находитесь, на каком символе этой линии и на каком символе файла в целом. Можно ещё нажать Alt+# (может различаться от дистрибутива, нужно смотреть в подсказке Ctrl+G) и слева появится нумерация строк. Но если мы закроем nano и заново откроем, то придётся заново включать нумерацию.

```
user@centos8:~  
File Edit View Search Terminal Help  
GNU nano 2.9.8 /home/user/.nanorc  
set linenumbers  
^G Get Help ^O Write Out ^W Where Is ^K Cut Text  
^X Exit ^R Read File ^\ Replace ^U Uncut Text
```

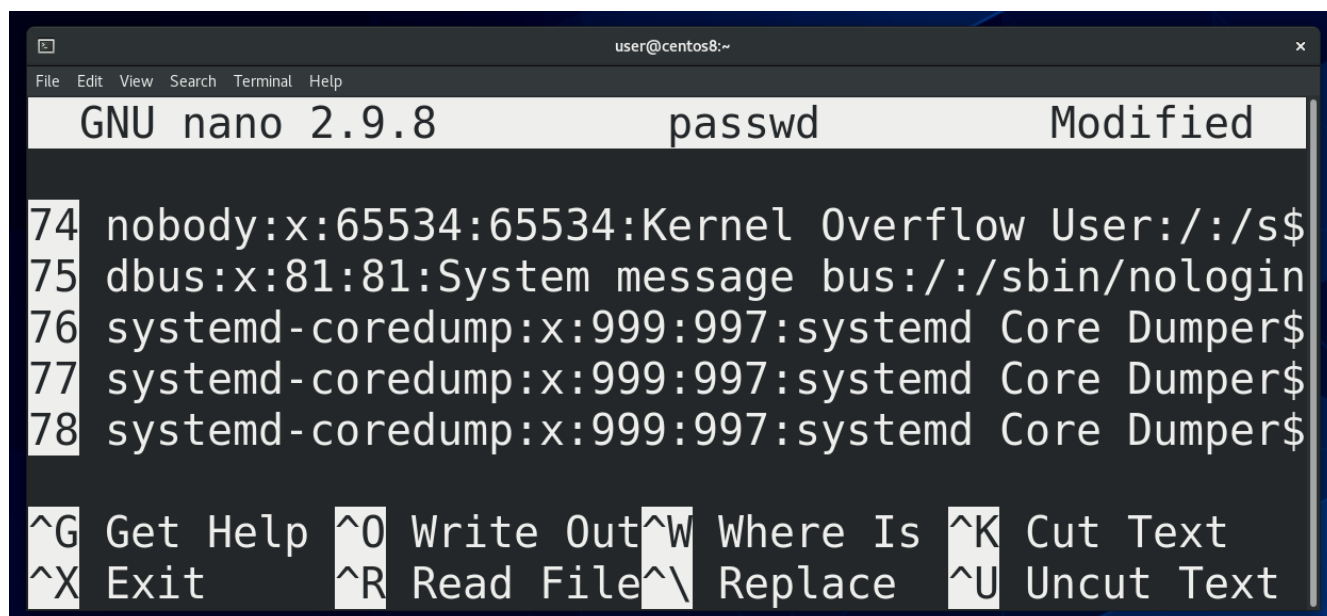
А давайте сделаем так, чтобы нумерация была всегда видна – для этого нужно подредактировать файл конфигурации nano – nanorc, который должен находиться в домашней директории пользователя и должен быть скрытым, то есть имя должно начинаться с точки. Пишем – nano ~/.nanorc. Я использую тильду слэш (~/), потому что не важно, где я нахожусь, тильда слэш всегда ведёт в домашнюю директорию, ну и название файла - .nanorc. Файл новый, потому что до этого мы никаких настроек nano не сохраняли. Пишем - set linenumbers - сохраняем и выходим. Открываем файл – nano passwd – и вот теперь по умолчанию нумерация включена.

```
user@centos8:~  
File Edit View Search Terminal Help  
GNU nano 2.9.8 passwd  
1 root:x:0:0:root:/root:/bin/bash  
2 bin:x:1:1:bin:/bin:/sbin/nologin  
3 daemon:x:2:2:daemon:/sbin:/sbin/nologin  
4 adm:x:3:4:adm:/var/adm:/sbin/nologin  
5 lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin  
Search: user  
^G Get Help M-C Case Sen M-B Backward M-J FullJstify  
^C Cancel M-R Regexp ^R Replace ^T Go To Line
```

Теперь попытаемся найти строчку нашего пользователя user – нажимаем Ctrl+W и появляется надпись Search: - пишем user. Вариантов может быть несколько. Чтобы перемещаться между вариантами, нажимаем Alt+W.

```
user@centos8:~  
File Edit View Search Terminal Help  
GNU nano 2.9.8 passwd  
1 root:x:0:0:root:/root:/bin/bash  
2 bin:x:1:1:bin:/bin:/sbin/nologin  
3 daemon:x:2:2:daemon:/sbin:/sbin/nologin  
4 adm:x:3:4:adm:/var/adm:/sbin/nologin  
5 lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin  
Enter line number, column number: 13 5  
^G Get Help ^Y First Line ^T Go To Text  
^C Cancel ^V Last Line
```

Теперь попытаемся перейти сразу на какую-то строчку – пишем Ctrl+W, а потом Ctrl+T – и появляется строка с предложением ввести номер строки и символа. Можно просто написать номер строки и Enter, либо номер строки и символа через пробел – 13 6.



```
user@centos8:~
File Edit View Search Terminal Help
GNU nano 2.9.8 passwd Modified
74 nobody:x:65534:65534:Kernel Overflow User:/:/s$
75 dbus:x:81:81:System message bus:/:/sbin/nologin
76 systemd-coredump:x:999:997:systemd Core Dumper$
77 systemd-coredump:x:999:997:systemd Core Dumper$
78 systemd-coredump:x:999:997:systemd Core Dumper$
^G Get Help ^O Write Out ^W Where Is ^K Cut Text
^X Exit ^R Read File ^\ Replace ^U Uncut Text
```

Часто бывает нужно вырезать или скопировать целую строку – нажимаем Ctrl+K, чтобы вырезать и Ctrl+U чтобы вставить. Можно переместиться в другое место и снова нажать Ctrl+U, чтобы вставить. Очень удобно, когда нужно несколько похожих строк. Можно разом вырезать несколько строк – нажимаем Ctrl+K несколько раз, а потом при Ctrl+U вставляются все вырезанные строки. Чтобы отменить последние изменения, нажимаем Alt+U.

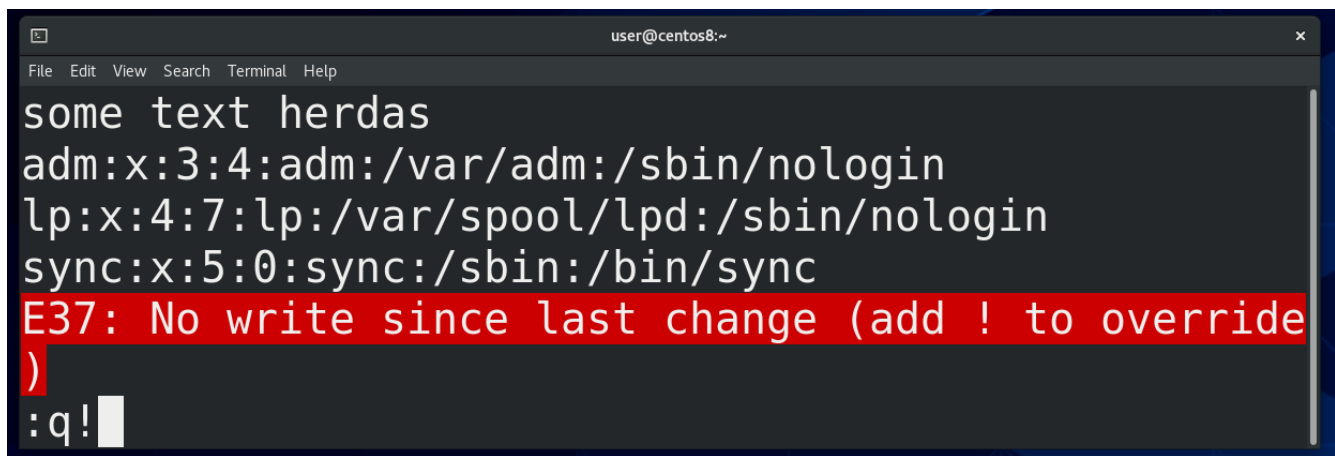
Ладно, не будем переусложнять. Для начала вышесказанных горячих клавиш для работы с nano вполне хватит. Некоторые другие клавиши мы рассмотрим, когда непосредственно начнём работать с файлами. Теперь перейдём к vi. Я не буду также детально рассматривать vi, только пройду по самому необходимому. Сам я vi не пользуюсь без необходимости, но для желающих в интернете миллион гайдов и даже игра, объясняющая, как работать с vi <https://vim-adventures.com/>.

```
user@centos8:~  
n:x:1:1:bin:/bin:/sbin/nologin  
daemon:x:2:2:daemon:/sbin:/sbin/nologin  
adm:x:3:4:adm:/var/adm:/sbin/nologin  
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin  
sync:x:5:0:sync:/sbin:/bin/sync  
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown  
halt:x:7:0:halt:/sbin:/sbin/halt  
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin  
operator:x:11:0:operator:/root:/sbin/nologin  
1,1 Top
```

Точно как и с nano, вы можете написать vi file, чтобы создать или изменить файл – vi passwd. В vi есть несколько режимов – командный режим, режим ввода и режим последней строки. Когда вы открываете vi, вы оказываетесь в командном режиме – в этом режиме вы не можете писать текст, но можете выполнять команды, например - x – для удаления символа или два раза d для удаления строки. Backspace в vi не работает. Но стоит учесть, что Centos вместо vi открывает vim, а в нём backspace работает (в режиме ввода).

```
user@centos8:~  
n:x:1:1:bin:/bin:/sbin/nologin  
daemon:x:2:2:daemon:/sbin:/sbin/nologin  
some text here  
adm:x:3:4:adm:/var/adm:/sbin/nologin  
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin  
sync:x:5:0:sync:/sbin:/bin/sync  
-- INSERT -- 3,15 Top
```

Чтобы перейти в режим ввода, можно использовать несколько клавиш - нажимаем i маленькое - внизу появляется надпись INSERT - и начинаем писать ровно там, где был курсор. Чтобы выйти из режима ввода, нажимаем Esc. I большое, то есть Shift+i – начинаем писать с начала строки. а маленькое – начинаем писать после курсора, A большое – в конце строки. o – добавляем строку ниже и начинаем в ней писать, O – строка сверху. Напомню, чтобы удалить текст, используем backspace, либо переходим в режим команд, то есть нажимаем Esc, наводим курсор куда нужно и нажимаем x. Если хотим сохранить это безобразие, в командном режиме нажимаем Shift+Z+Z.



```
user@centos8:~  
File Edit View Search Terminal Help  
some text herdas  
adm:x:3:4:adm:/var/adm:/sbin/nologin  
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin  
sync:x:5:0:sync:/sbin:/bin/sync  
E37: No write since last change (add ! to override)  
):q!
```

Ну и третий режим – режим последней строки. Чтобы перейти в него, нужно в командном режиме написать двоеточие (:) - внизу появится двоеточие. Тут вообще много всяких команд можно ввести, но нас интересуют основные - как сохранить и как выйти. И так, если никаких изменений нет – то пишем q, то есть чтобы стало :q и нажимаем Enter. Если вы сделали какие-то изменения, то есть, нажали i, ввели какой-то текст, потом нажали Esc и двоеточие, то при попытке написать q и Enter внизу появится ошибка с подсказкой, что нужно добавить восклицательный знак. Пишем :q! и Enter – никакие изменения не сохраняются. Если хотим сохранить изменения – пишем w и enter и изменения сохраняются. Можем разом сохранить изменения и выйти, как при использовании Shift+Z+Z - для этого пишем :wq .

Не смотря на все эти махинации, vi, а точнее vim, очень любят в народе. Для задач администрирования функционала папо вполне хватает, но каждый сам решает, с каким редактором ему работать. И тем не менее, обязательно научитесь работать с vi, хотя бы базово – открыть файл, изменить какой-то текст и сохранить, либо не сохранить.