

# Основы системного администрирования и сетевых технологий

---

УРОК №6

## Текстовый редактор vi

Наиболее популярным редактором для работы с текстовыми файлами в Linux является vi.

На самом деле существует ряд программ, либо в точности повторяющих поведение Vi (например, nVi), либо очень похожих на Vi, но со значительно расширенными возможностями (например, Vim). Более того, когда пользователь Linux набирает в командной строке Vi, скорее всего, будет запущен облегченный вариант Vim, которая настроена таким образом, чтобы максимально воспроизводить поведение классического Vim. Чтобы вызвать Vim в полнофункциональном режиме, необходимо набрать команду Vim.

Чтобы открыть для редактирования уже существующий файл, путь к этому файлу нужно указать в качестве параметра при вызове Vi:

vi путь\_к\_файлу.

Если вызвать редактор без параметров, откроется пустой *буфер*.

Для отображения текста и работы с ним используется весь экран терминала, за исключением нижней строки, которая предназначена для диалога с пользователем.

При работе с консольным текстовым редактором надо как-то различать работу непосредственно с текстом и команды, которые необходимо давать редактору. Чтобы различать эти сущности в Vi используется понятие *режима*:

– **в командном режиме** – вводимый текст - это команда редактору, нажатие клавиши считается командой и немедленно выполняется (но никогда не отображаются). Командный режим используется для перемещения курсора, операций вырезания и вставки и прочих действий, не связанных с набором текста;

## Текстовый редактор vi

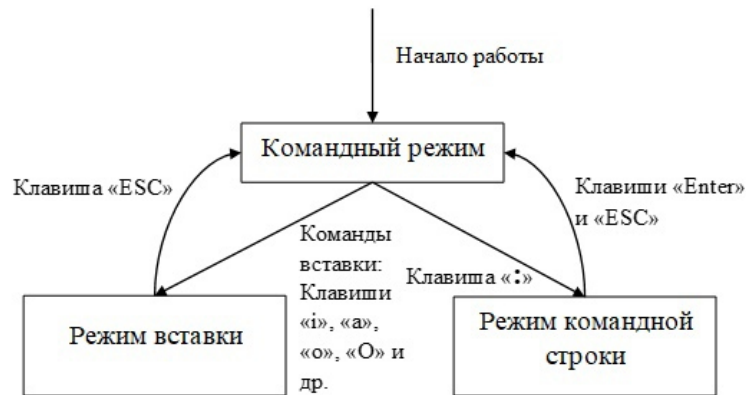


Рисунок 1. Диаграмма переключения режимов Vi

– **в режиме вставки** – нажатие клавиши - вставка соответствующего символа;

– **в режиме командной строки** – вводимый текст это длинная команда, для завершения которой используется клавиша Enter.

У Vi есть своя командная строка. Чтобы ее вызвать нужно в командном режиме ввести «:». В результате в последней строке появятся символы «..» – приглашение командной строки.

Важно уметь получать помощь по Vi. Для этого необходимо:

1) перевести Vi в режим командной строки, нажав «:»;

2) ввести команду help и нажать клавишу Enter.

Для того, чтобы закрыть справочную страницу, нужно ввести команду :q.

Чтобы ввести текст, необходимо перейти из командного режима в режим вставки, нажав клавишу «i» (insert – вставка).

После этого в последней строке появится сообщение о том, что редактор находится в режиме вставки «--INSERT--».

В режиме вставки можно вводить текст, завершая строку нажатием клавиши Enter.

Чтобы перейти из режима вставки в командный режим, необходимо нажать клавишу «ESC».

Диаграмма переключения режимов Vi показана на рисунке 1.

Для переключения из командного режима в режим вставки существует много способов, а для выхода из режима вставки только один – нажатие клавиши «ESC».

## Команды позиционирования курсора

Команды позиционирования курсора, которые могут комбинироваться с другими командами, приведены в табл. 1.  
Таблица 1. Команды позиционирования курсора, которые могут комбинироваться с другими командами

Команда	Описание
j	Перемещение курсора на одну строку вниз
k	Перемещение курсора на одну строку вверх
h	Перемещение курсора на один столбец влево
l	Перемещение курсора на один столбец вправо
Enter	Перемещение курсора на одну строку вниз
G	Переход на строку. Если не указано количество повторений, осуществляется переход в конец файла. Если количество повторений указано, то осуществляется переход на соответствующую строку. Например, при вводе 50G осуществляется переход на 50 строку файла.
+	Перемещение курсора на одну строку вниз с позиционированием его возле первого непустого символа строки.
-	Перемещение курсора на одну строку вверх с позиционированием его возле первого непустого символа строки.
%	Переход к соответствующей скобке. Курсор позиционируется у символа скобки, например (), {} или [], в противном случае позиционирование не осуществляется
[[	Двухсимвольная команда, которая перемещает курсор обратно к первой фигурной скобке { в первом столбце. Полезна при перемещениями между функциями.
]]	Аналог [[, за тем исключением, что перемещение осуществляется вперед.

'{mark}	Переход на строку, помеченную определенной отметкой при помощи команды m. Отметки определяются пользователем с помощью команды m и могут быть любыми символами.
``	Возврат на последнюю строку, с которой осуществлялся переход или поиск.
/ {expr}	Осуществление поиска в прямом направлении. Если нажать клавишу /, будет выведено приглашение ввести регулярное выражение для поиска. Поиск начинается после нажатия клавиши Enter. Он начинается с текущей строки и продолжается до первого совпадения.
? {expr}	Осуществление поиска в обратном направлении. Команда является аналогом /, за исключением того, что поиск осуществляется от текущей строки к началу файла.
n	Повторение последнего поиска в том же направлении, начиная с текущей позиции курсора.
N	Повторение последнего поиска в противоположном направлении, начиная с текущей позиции курсора.
w	Перемещение курсора на одно слово вперед
b	Перемещение курсора на одно слово назад

Команды позиционирования курсора, которые не могут комбинироваться с другими командами, приведены в табл. 2.  
Таблица 2. Команды позиционирования курсора, которые не могут комбинироваться с другими командами

Команда	Описание
Ctrl+F	Перемещение на одну экранную страницу вперед
Ctrl+B	Перемещение на одну экранную страницу назад
Ctrl+]	Переход к тегу, на который указывает курсор. Используется с ctags.
Ctrl+T	Возврат с предыдущего перехода к тегу (аналог кнопки Back в вэб-браузере).

### Команды вставки

Существует множество способов перехода в режим вставки, но в большинстве случаев оказывается вполне достаточным использование команды `i`. Команда `i` и другие команды приведены в табл. 3.

Таблица 3. Команды вставки

Команда	Описание
<code>i</code>	Переход в режим вставки. Курсор перед символом.
<code>a</code>	Переход в режим вставки. Курсор после символа.
<code>I</code>	Переход в режим вставки. Курсор перед первым непустым символом строки.
<code>A</code>	Переход в режим вставки. Курсор после последнего непустого символа строки.
<code>o</code>	Переход в режим вставки, начиная с новой строки под текущей позицией курсора.
<code>O</code>	Переход в режим вставки, начиная с новой строки над текущей позицией курсора.

### Команды изменения

Часто используемые команды изменения приведены в табл. 4  
Таблица 4. Команды изменения

Команда	Описание
<code>C</code>	Удаление всего, что следует после курсора, до конца строки, затем осуществляется переход в режим вставки.
<code>c{motion}</code>	Удаление текста, начиная с символа после курсора в соответствии с аргументом <code>motion</code> , затем происходит переход в режим вставки.
<code>{N}s</code>	Изменение (замена) следующих <code>N</code> символов, начиная с текущего положения курсора. Команда аналогична команде <code>c</code> , за тем исключением, что вместо команды перемещения здесь указывается точное количество символов, например <code>5s</code> .
<code>S</code>	Изменение (замена) текущей строки полностью. Удаляет весь текст текущей строки и выполняет переход в режим вставки.

## Команды изменения

Шаблоны использования команд изменения показаны в табл. 5.

Таблица 5. Шаблоны использования команд изменения

Шаблон	Описание
2cw	Удаление двух слов и переход в режим вставки; альтернативный вариант – удаление двух следующих слов.
cta	Удаление всего вплоть до следующего появления буквы «а» с последующим переходом в режим вставки. Является комбинацией команды с и команды перемещения ta.
5cta	Удаление всего вплоть до пятого появления буквы «а» с последующим переходом в режим вставки.
5S	Удаление текущей строки и последующих 4 строк с последующим переходом в режим вставки.

## Прочие команды

В табл. 6 приведены другие команды, которые часто используются при практической работе.

Таблица 6. Другие часто используемые команды

Команда	Описание
U	Отмена редактирования, касающегося последней операции изменения или вставки.
Ctrl+R	Возврат последнего отмененного редактирования.
m{letter}	Устанавливает закладку на текущей строке посредством задания буквы. Буква может быть любой, но в нижнем регистре в соответствии с текущей локализацией.
.	Повтор последней операции изменения или вставки.
Ctrl+L	Обновление экрана.
zt	Обновление экрана с размещением текущей строки и курсора в верхней части экрана.
zz	Обновление экрана с размещением текущей строки и курсора в средней части экрана.
zb	Обновление экрана с размещением текущей строки и курсора в нижней части экрана.

## Команды вырезания, вставки и удаления

Эти команды отличаются от команд изменения тем, что они не используются в режиме вставки. При вводе этих команд необходимо находиться в командном режиме. Как и большинство команд, эти команды могут повторяться, то есть, если ввести 5р, то вставка в редактор содержимого буфера будет выполнена 5 раз.

В табл. 7 приведены команды вырезания, вставки и удаления. Таблица 7. Команды вырезания, вставки и удаления

Команда	Описание
D	Удаление всего, начиная с текущего положения курсора до конца строки. Данные сохраняются в буфере по умолчанию.
d{motion}	Удаление определенного количества символов, начиная с текущего положения курсора. Количество символов определяется аргументом motion. Для удаления текущей строки надо ввести dd. Данные сохраняются в буфере по умолчанию.
y{motion}	Копирование определенного количества символов в регистр. Количество символов определяется аргументом motion. Для копирования текущей строки в буфер обмена надо ввести уу.
р	Вставка (помещение) в буфер символов, начиная с символа, следующего за символом возле курсора.
Р	Вставка (помещение) в буфер символов, начиная с символа, находящегося перед символом возле курсора.

Команды вставки и удаления могут принимать один аргумент с меткой motion, который сообщает редактору, какой фрагмент текста необходимо удалить или скопировать. Аргументом motion может быть любая из команд перемещения курсора, а также следующие дополнительные команды перемещения, приведенные в табл. 8.



## Команды вырезания, вставки и удаления

Таблица 8. Дополнительные команды перемещения

Команда	Описание
f{char}	Позиционирует курсор у первого столбца справа, что соответствует указанному символу.
t{char}	Аналог f, за исключением того, что курсор позиционируется на один столбец левее соответствующего символа.
F{char}	Позиционирует курсор у первого столбца слева, что соответствует указанному символу
T{char}	Аналог F, за тем исключением, что курсор позиционируется на один столбец правее соответствующего символа.

В следующей табл. 9 приведен ряд базовых шаблонов для команд d и u, которые демонстрируют, как эти команды можно комбинировать с командами перемещения.

Таблица 9. Базовые шаблоны для команд d и u

Команда	Описание
dfa	Удаление символов, начиная с текущего положения курсора вправо до первой буквы «а», которая также удаляется. Мнемоническое правило: удалять все символы, пока не встретится буква «а».
dta	Удаление символов, начиная с текущего положения курсора вправо до первой буквы «а», которая не будет удаляться. Мнемоническое правило: удалять все символы, вплоть до буквы «а».
5yta	Копирование символов, начиная с текущего положения курсора вплоть до пятого появления буквы «а».
yy4p	Копирование текущей строки и вставка в буфер четырех дополнительных копий. В данном случае используются две команды: 1) yy – вставка в буфер обмена текущей строки и 2) 4p – вставка в буфер четырех копий.

Команда	Описание
dn	Удаление символов, начиная с текущего положения курсора до первого соответствия самого последнего поиска. Здесь команда d комбинируется с командой перемещения n.
d`a	Удаление символов, начиная с текущего положения курсора вплоть до позиции с отметкой «a». Команда d комбинируется с командой `a в качестве команды перемещения.
yG	Копирование всех строк, начиная с текущей строки до конца файла. В данном случае команда комбинируется с командой перемещения G, которая позиционирует курсор в конце файла.
y50G	Копирование всех строк, находящихся между текущей и 50-строкой. Команда y комбинируется с командой перемещения 50G.
d5i	Удаление следующих пяти символов, начиная с текущего положения курсора. Команда d комбинируется с командой перемещения 5i.
5dd	Удаление текущей строки и четырех следующих строк. Команде dd предшествует количество повторений, равное 5.

### Режим командной строки

В режим командной строки можно перейти, нажав клавишу «:». Редактор будет находиться в этом режиме до тех пор, пока не будет нажата клавиша Enter, после чего редактор перейдет в командный режим.

Все команды командного режима имеют одинаковую базовую форму: сначала идет необязательный номер строки или диапазон номеров строк, затем следует команда:

`:[firstline][,lastline] command`

Если номера строк не указаны, команда будет применена к текущей строке.

Например, чтобы удалить строки с 10 по 20 включительно, нужно воспользоваться командой

`:10,20d`

**Режим командной строки**

Для ввода определенных номеров строк можно использовать горячие клавиши, некоторые из которых приведены в следующей табл. 10.

Таблица 10. Клавиши для ввода определенных номеров строк

Символ	Номер строки
.	Номер текущей строки
\$	Номер последней строки в файле
%	Все содержимое файла, аналог 1,\$
`a	Местоположение тега a. В командном режиме теги устанавливаются с помощью команды m.
/ {expr} /	Следующая строка, соответствующая регулярному выражению
? {expr} ?	Предыдущая строка, соответствующая регулярному выражению

Чаще всего при работе в режиме командной строки используются следующие команды, приведенные в табл. 11.

Таблица 11. Часто используемые команды в режиме командной строки

Команда	Краткая форма	Описание
write	:w{filename}	Запись содержимого экрана в файл. Если имя файла не указано, то обновляется текущий файл на диске. Используйте :w! Для того, чтобы записывать в, помеченный «только для чтения».
quit	:q	Выход из текстового редактора. Команда не сработает, если содержимое файла не было сохранено. Чтобы выйти из редактора без сохранения, используется команда :q!

Команда	Краткая форма	Описание
xit	:x	Выход из текстового файла и сохранение несохраненных файлов. Чтобы сохранить в файл, помеченный «только для чтения», используется команда :x!
edit	:e{filename}	Открытие указанного файла в новом буфере. Текущий файл остается открытым. Если имя файла не указано, заново открывается текущий файл. Если вы решили не сохранять изменения и хотите заново открыть файл, используйте команду :e!
delete	:d	Удаление текущей строки или диапазона строк
map	map{a}{b}	Переназначение клавиш, используемых в командном режиме
set	:set{argument}	Изменение настроек по умолчанию для текстового редактора
help	:help	Вход в справочную систему. Справочная система организована в виде гиперссылок с использованием тэгов

### Поиск и замена

Базовой командой для поиска и замены в командном режиме является команда `substitute`, которая может быть введена в короткой форме `s`. Базовая команда `substitute` выглядит следующим образом:

```
:s/search/replacement/flags
```

Единственным необходимым аргументом является строка поиска. Параметры `replacement` и `flags` являются опциональными. Все аргументы должны разделяться знаком прямого слеша «/», что не совсем удобно при работе с именами файлов. Редактор `vi` требует, чтобы все прямые слешы заменялись знаками обратного слеша «\». Поэтому команда замены с указанным путем будет выглядеть следующим образом:

```
:s/\usr\bin\file1\usr\bin\file2\//
```

## Модификация настроек vi

Настройки vi хранятся в домашнем каталоге пользователя в файле .exrc. Если используется vim, то его настройки хранятся в файле .vimrc.

Многие настройки vi можно устанавливать в режиме командной строки с помощью команды :set.

В табл. 12 содержится ряд важных настроек, которые можно модифицировать.

Таблица 12. Настройки, которые можно модифицировать

Команда	Краткая форма	Пример	Описание
tabstop	ts	set ts=4	Определение количества столбцов в табуляции (по умолчанию 8). Это влияет на то, как отображается текст, содержащий табуляции
shiftwidth	sw	set sw=4	Определяет количество столбцов, на которое осуществляется сдвиг при использовании команд отступа (по умолчанию 8). Этот параметр не зависит от настройки tabstop.
autoindent	ai	set ai	Включение или выключение автоматического отступа; ai - включен, noai - выключен.
expandtabs	et	set et	Не вставлять табуляции, вместо этого использовать число пробелов, определяемое настройкой tabstop. По умолчанию используются табуляции.
wrapscan	ws	set ws	Изменение порядка поиска; если данная настройка активирована (по умолчанию - активирована), при прямом поиске будут выявляться соответствия в предыдущих строках, а при обратном поиске - в последующих строках. Если настройка выключена, прямой поиск осуществляется с текущей строки до конца файла. Обратный поиск начинается с текущей позиции и продолжается до начала файла.

# Основы системного администрирования и сетевых технологий

---