

[Další Předchozí Obsah](#)

4. Základy editoru vi

vi je standardním Unixovým programem pro editaci textů. Jeho zvládnutí je pro systémové administrátory nezbytné. Existuje několik klonů či verzí programu vi: vi samotný, dále `elvis`, `vi10`, `a vim`. Aspoň jeden z nich se vyskytuje snad v každé verzi Unixu, a tedy i Linuxu. Všechny jeho verze obsahují stejné základní vlastnosti a sadu příkazů, takže zvládnutí jedné verze by mělo usnadnit zvládnutí ostatních.

vi nabízí množství užitečných pomůcek jako je zvýrazňování syntaxe, formátování kódu, výkonné mechanismy pro vyhledávání a nahrazování, makra a další. Tyto vlastnosti jej činí obzvláště atraktivním pro programátory, webové vývojáře a podobně. Systémoví administrátoři ocení automatizaci a integraci s jakýmkoli dostupným shellem.

4.1 Spouštění vi

vi může být spouštěn z příkazové řádky několika způsoby. Tím nejjednodušším je:

```
$ vi
```

vi nastartuje s prázdným bufferem. V tomto momentě uvidíte nejspíše prázdnou obrazovku. Je to *příkazový mód* a čeká se na vás, že řeknete, co se má dělat. Abyste vi ukončili, napište:

```
:q
```

Pokud nebyly provedeny žádné změny v editovaném souboru, vi se bezprostředně ukončí. Pokud změny učiněny byly, bude vás vi varovat a současně vás poučí, jak toto varování přebít a problém ignorovat (napište `:q !`).

Také můžete spouštět vi s nějakým už existujícím souborem. Například soubor `/etc/resolv.conf` můžete otevřít takto:

```
$ vi /etc/resolv.conf
```

Dokonce můžete vi spustit tak, aby se zadaný soubor otevřel na nějakém konkrétním řádku. Například můžete spustit vi na 47. řádku souboru `/usr/src/linux/init/main.c`. Uděláte to takto:

```
vi +47 /usr/src/linux/init/main.c
```

vi zobrazí zadaný soubor a umístí kurzor na příslušný řádek. Pokud by číslo řádku bylo vyšší, než jeich počet v souboru, umístí se kurzor na poslední řádek. Tato volba je obzvláště užitečná pro programátory, neboť jim umožňuje skočit rovnu na řádek, kde se objevila chyba.

4.2 Režim práce - módy

Editor vi pracuje v různých módech, které se používají k provádění různých úkolů. Když nastartujete vi, dostanete se do *příkazového módu* (command mode). V tomto režimu můžete zadávat různé příkazy pro manipulaci s textem, pohybovat se v textu, ukládat, ukončovat a měnit módy (and mode). Editování textu se provádí v *módu vkládání* (insert mode). Mezi jednotlivými módy se můžete přepínat snadno pomocí různých

1 z 5

7.11.2006 13:54

Skolení UNIX: Základy editoru vi

http://www.kiv.zcu.cz/~simekm/skoleni/skoleni-unix-vi.html

Někdy možná budete potřebovat uložit soubor, který je označen jako "read-only" (jen pro čtení). Dosáhnete toho přidáním vykřičníku za příkaz k uložení:

```
!w!
```

```
!wq!
```

Ukončování vi

Jeden ze způsobů jak ukončit vi je příkaz `!wq`, který uloží aktuální buffer a skončí. Nebo můžete ukončit editor bez ukládání pomocí `:q` či `:q!`.

4.4 vi klávesy

Tato část je stručnou referenční příručkou nepoužívanějších příkazů vi. O některých z nich jsme psali už dříve v této kapitole, zatímco jiné budou nové.

Přesouvání

Operace	Klávesa
Vlevo, dolů, nahoru, doprava	<code>h</code> , <code>j</code> , <code>k</code> , <code>l</code>
Na konec řádku	<code>\$</code>
Na začátek řádku	<code>^</code>
Na konec souboru	<code>g</code>
Na začátek souboru	<code>1</code>
Na řádek 47	<code>:47</code>

Editování

Operace	Klávesa
Vyjmutí řádku	<code>dd</code>
Vyjmutí pěti řádků	<code>5dd</code>
Přepsání znaku	<code>r</code>
Vyjmutí znaku	<code>x</code>
Vyjmutí deseti znaků	<code>10x</code>
Odvolení poslední akce (undo)	<code>u</code>
Spojení aktuálního a následujících řádků	<code>J</code>

Vyhledávání

Operace	Klávesy
Vyhledání asdf	<code>/asdf</code>
Zpětné hledání asdf	<code>?asdf</code>
Opakování posledního hledání vpřed	<code>/</code>
Opakování posledního hledání dozadu	<code>?</code>

Ukládání a ukončování

4 z 5

7.11.2006 13:54

Skolení UNIX: Základy editoru vi

http://www.kiv.zcu.cz/~simekm/skoleni/skoleni-unix-vi.html

kláves, které popíšeme dále.

Příkazový mód (Command Mode)

Na počátku se nacházíte v příkazovém módu. V tomto módu nemůžete přímo vkládat ani upravovat text. Nicméně můžete s textem různě manipulovat, vyhledávat, ukončovat, ukládat, načítat nové soubory atd. To co jsme si teď řekli berte jen jako přibližnutí k příkazovému módu.

Asi nepoužívanějším příkazem v příkazovém módu je přepnutí do *módu vkládání*. Toho se dosáhne stiskem klávesy `i`. Kurzor změní tvar a dole na obrazovce je zobrazen nápis `-- INSERT --` (to se nemusí dít v každém klonu editoru vi). Od této chvíle jsou všechny stisky kláves vkládány do aktuálního bufferu a jsou zobrazovány na obrazovce. K návratu do příkazového módu stisknete klávesu `escape`.

V příkazovém módu se můžete také pohybovat po souboru. Na některých systémech k tomu můžete použít kláves se šipkami. Na jiných si budete muset používat tradičnější klávesy `h`,`j`,`k`,`l`. Zde je jednoduchý seznam těchto kláves a jejich význam:

```
h      přesun o jeden znak vlevo
j      přesun o jeden znak dolů
k      přesun o jeden znak nahoru
l      přesun o jeden znak vpravo
```

Jednoduše stisknete některou z těchto kláves. Jak uvidíte později, tyto klávesy je možné kombinovat s čísly a pohyb je pak mnohem rychlejší.

Mnoho příkazů, které používáme v příkazovém módu, začíná dvojtečkou. Například pro ukončení programu vi zadáte příkaz `:q`. Dvojtečka indikuje, že se jedná o příkaz, zatímco `q` říká editoru, aby se ukončil (`quit`). Dalšími příkazy jsou (volitelně) číslo následované písmenem. Těmito příkazy dvojtečka nepředchází a jsou obecně používány pro manipulaci s textem.

Například vymazání řádku se provádí stiskem `dd`. Tak vymažete řádek, ve kterém je kurzor. Zadáním příkazu `4dd` řeknete editoru, aby smazal čtyři řádky (aktuální a ještě tři pod ním). Obecně tím číslem říkáme, kolikrát se má zadání příkaz zopakovat.

Čísla můžete kombinovat i s klávesami pro pohyb kurzoru a přesouvat se tak o větší počet pozic nahoru. Například `10x` vás přesune o deset řádků výše.

Příkazový mód lze rovněž využít pro vyjímání a vkládání textu a načtení jiných souborů do aktuálního bufferu. Výběr textu pro kopírování provádíme klávesou `y` (`y`=`yank`). Výběr celé aktuální řádky dosáhnete stiskem `yy`, čemuž může předcházet ještě číslo ke zkopírování více řádek najednou. Potom se přesunete do místa, kam se má vybraný text vložit a stisknete `p` (`p`=`paste`). Text se vloží za aktuální řádek.

A ještě si řekneme něco o vyhledávání. Příkazový mód nabízí jak prosté vyhledání textu, tak velmi složité příkazy pro vyhledávání a nahrazování, využívající skvělých možností regularních výrazů. Jednoduché hledání vyvoláme stiskem klávesy `/`, následované textem, který chceme najít. Editor pak hledá od pozice kurzoru dál směrem ke konci textu a zastaví se v místě, kde hledaný text najde. Přičemž nemusí jít o shodu celých slov, ale jen části. Například vyhledávání slova "lom" najde nejen samostatné slovo "lom", ale i slova "zlom", "plomba", "lomitko", ap. Je to proto, že všechna ta slova v sobě kombinaci písmen "lom" obsahují.

Poté, co vi našel první výskyt hledané fráze, můžete vyvolat další její hledání už pouhým stiskem `/` a `enteru`. Frázi znovu zapisovat nemusíte. Můžete vyhledávat i opačným směrem. Uděláte to tak, že místo lomitka použijete klávesu `?`. Například, chcete-li, aby se "lom" hledalo směrem k začátku textu, napište `?lom`.

Mód vkládání (Insert Mode)

Vkládání a přepisování textu provádíme v módu vkládání. Jak už jsme se zmínili, do módu vkládání se z

2 z 5

7.11.2006 13:54

Skolení UNIX: Základy editoru vi

http://www.kiv.zcu.cz/~simekm/skoleni/skoleni-unix-vi.html

Operace	Klávesa
Ukončení -Quit	<code>:q</code>
Ukončení bez ukládání	<code>:q!</code>
Uložit a ukončit	<code>:wq</code>
Uložit, ale nekončit	<code>:w</code>
Znovu načíst otevřený soubor	<code>o!</code>
Zaplat buffer do souboru	<code>!w asdf</code>
Otevřít soubor hojazz	<code>o hojazz</code>
Načtení souboru asdf do bufferu	<code>r asdf</code>
Načtení příkazu ls do bufferu	<code>r !ls</code>

Zdroje

- <http://www.kit.vstb.cz/~satrapa/docs/vim/> vim - skutečný editor textů
- <http://www.abclinuxu.cz/clanky/ViewRelation?relationId=24013> Editor VIM prakticky
- http://www-hep2.fzu.cz/computing/informace/subsection3_6_1.html#SECTION00061000000000000000 Stránkový editor vi
- <http://www.net.cz/clanek/708> Jak si přizpůsobit Vim - seriál
- http://jgu.cz/wraith/P000042/s_vim/o_vimu.html Editor Vim

[Další Předchozí Obsah](#)

5 z 5

Skolení UNIX: Základy editoru vi

http://www.kiv.zcu.cz/~simekm/skoleni/skoleni-unix-vi.html

příkazového módu přepnete stiskem klávesy `i`. Potom vše, co mačkáte na klávesnici se vkládá do aktuálního bufferu. Mód vkládání ukončíte stiskem klávesy `Esc`. Tím se vrátíte do příkazového módu.

Přepisování textu dosáhnete více způsoby. V příkazovém módu vám stisk klávesy `=` umožní nahradit jeden znak, který se nachází pod kurzorem. Pak jen stisknete nový znak a ten se objeví místo původního a je to. Pak vás to hned přepne zpět do příkazového módu. Stiskem `=` se přepnete do přepisovacího módu. Ten vám umožní přepisovat tolik textu, kolik jen chcete. Do příkazového módu se vrátíte jako obvykle stiskem klávesy `Escape`.

Některé verze editoru vi vám nabízejí možnost přepínat se mezi módy vkládání a přepisování přímo. Stisknete klávesu `insert`. To vás přenese do módu vkládání. Jakmile jste v módu vkládání fungují další stisky klávesy `insert` jako přepínač mezi módy vkládání a přepisování.

4.3 Práce se soubory

Otevírání souborů

Soubor můžete otevřít dvěma způsoby. Jednak při spouštění editoru z příkazové řádky můžete zadat i jméno souboru, který pak bude po nastartování otevřen. Nebo, když už máte vi spuštěný, použitím příkazu `o` v příkazovém módu. Například `/etc/lilo.conf` otevřete takto:

```
o /etc/lilo.conf
```

Soubor se načte do bufferu a můžete s ním pracovat. Nebyl-li buffer před otevřením souboru prázdný (měl jste tu otevřený jiný soubor), bude stávající soubor v bufferu nahrazen nově otevřeným. Nebyl-li starý soubor před otevíráním nového uložen, bude vi protestovat a otevření nemožno (poslední provedené úpravy ve stávajícím souboru by se ztratily). Pokud vám to nevádí, řeknete to editoru tak, že pro otevření použijte upravený příkaz `o!`. Obecně vždycky vykřičníkem můžete přebít případné varovné hlásky editoru.

Některé klony editoru vi (například `vi+n`) umožňují mít otevreno více bufferů současně. Pak například pro otevření souboru `09-vi.sgml` v novém bufferu použijete příkaz:

```
!split 09-vi.sgml
```

Obrazovka se pak vodorovně rozptílí a nový soubor bude zobrazen v její horní polovině, zatímco ten stávající bude v dolní polovině. Různými příkazy pak můžete toto rozdělení obrazovky měnit. Znalciům editoru Emacs ty příkazy budou povědomé. Nejlepší způsob jak se o těchto příkazech něco dovědět je prostudovat manuálovou stránku vašeho vi klonu. Poznamenejme, že mnoho klonů myšlenku rozdělení obrazovky nepodporují, takže ji nebudete moci využít.

Ukládání souborů

vi nabízí několik způsobů, jak uložit soubor. Chcete-li uložit obsah aktuálního bufferu do souboru `randomness`, napište:

```
!w randomness
```

Jakmile máte soubor jednou uložený, jeho opětovné uložení provedete už pouhým zápsáním `!w`. Tím uložíte veškeré provedené změny. Po uložení souboru se vi vrátí do příkazového módu. Chcete-li nejen uložit soubor, ale rovněž ukončit práci s editorem (což je dost častá kombinace), můžete zapsat `!wq`. To říká editoru vi, aby uložil soubor a ukončil svou činnost.

3 z 5

7.11.2006 13:54