

# Soubory a adresáře

SOUBORY A ADRESÁŘE	
<b>pwd</b>	vypíše cestu k aktuálnímu (pracovnímu) adresáři
<b>cd   cd ~   cd \$HOME</b>	vstoupí do domovského adresáře přihlášeného uživatele
<b>cd &lt;adresar&gt;</b>	vstoupí do daného adresáře; cesta se uvádí buď absolutní (začíná znakem „/“) nebo relativní (určená přímo názvem adresáře - pokud jde o podadresář pracovního adresáře, či znakem „..“ nebo „-“, který představuje adresář nadřazený či předchozí)
<b>tree [&lt;adresar ...&gt;]</b>	vypíše obsah pracovního či zadaného adresáře v stromové struktuře, <b>-a</b> včetně skrytých souborů, <b>-d</b> jen adresáře, <b>-L &lt;n&gt;</b> max. do dané úrovně adresářového stromu, <b>-p</b> přístupová práva, <b>-u</b> vlastníka souboru, <b>-g</b> skupinu vlastníků souboru, <b>-s</b> velikost souboru v B \$ tree -a   tail -1 (vypíše celkový počet adresářů a souborů v pracovním adresáři)
<b>ls   dir [&lt;soubor ...   adresar ...&gt;]</b>	vypíše název souboru nebo obsah adresáře podle abecedy, <b>-a</b> včetně skrytých souborů, <b>-d</b> samotný adresář bez obsahu, <b>-i</b> číslo i-uzlu (i-nodu), <b>-l</b> podrobný výpis (typ souboru („-“ = soubor, „d“ = adresář, „l“ = symbolický odkaz, „p“ = pojmenovaná roura, „b“ = blokové zařízení, „c“ = znakové zařízení), přístupová práva, počet pevných odkazů, vlastník, skupina, velikost v B, čas poslední změny, název; znak „+“ na konci oprávnění znamená, že jsou nastavena práva ACL, znak „.“ sděluje, že jsou nastaveny speciální atributy), <b>-n</b> obdoba volby „-l“, ale vypíše numerická ID uživatelů a skupin, <b>-h</b> s volbou „-l“ vypíše velikosti v čitelném formátu, <b>-r</b> v opačném pořadí, <b>-c</b> podle poslední změny atributů souboru (přístupová práva, počet pevných odkazů, vlastník, skupina, velikost v B, čas poslední změny, název), <b>-t</b> podle poslední změny obsahu souboru, <b>-u</b> podle posledního přístupu, <b>-A</b> nezobrazí implicitní „.“ a „..“, <b>-F</b> zobrazí typ položky („*“ = spustitelný soubor, „@“ = symbolický odkaz, „/“ = adresář, „=“ = socket, „ “ = pojmenovaná roura), <b>-R</b> rekurzivně, <b>-S</b> podle velikosti, <b>-U</b> netřídí obsah podle abecedy, <b>-X</b> podle přípony, <b>-Z</b> vypíše bezpečnostní kontext SELinuxu, <b>-l</b> vypíše jeden soubor na řádek; bez argumentu vypíše obsah pracovního adresáře \$ ls -latr (vypíše obsah pracovního adresáře s naposledy změněnými soubory a adresáři na konci výpisu)
<b>stat &lt;soubor ...   adresar ...&gt;</b>	vypíše podrobné atributy souboru či adresáře (název, velikost v B, počet bloků, typ souboru, číslo i-uzlu, přístupová práva, vlastníka, skupinu, bezpečnostní kontext SELinuxu, čas posledního přístupu, čas poslední změny atributů a čas poslední změny obsahu souboru), <b>-c &lt;format&gt;</b> zobrazí výstup v daném formátu ( <b>%a</b> přístupová práva v číselném vyjádření, <b>%A</b> přístupová práva v symbolickém vyjádření, <b>%b</b> počet bloků, <b>%F</b> typ souboru, <b>%g</b> ID skupiny, <b>%G</b> jméno skupiny, <b>%i</b> číslo i-uzlu, <b>%n</b> jméno souboru, <b>%s</b> velikost v B, <b>%u</b> ID vlastníka, <b>%U</b> jméno vlastníka, <b>%x</b> čas posledního přístupu, <b>%y</b> čas poslední změny atributů souboru, <b>%z</b> čas poslední změny obsahu souboru), <b>-f</b> zobrazí informace o souborovém systému \$ stat -c "%U %G" soubor (vypíše vlastníka a skupinu vlastníků souboru)
<b>du [&lt;soubor ...   adresar ...&gt;]</b>	vypíše v kB velikost diskového prostoru, kterou daný soubor či adresář zabírá, <b>-a</b> vypíše velikost všech souborů a adresářů rekurzivně, <b>-b</b> vypíše velikost v B, <b>-m</b> vypíše velikost v MB, <b>-h</b> vypíše velikost v čitelném formátu, <b>-s</b> celkový součet bez podrobností; bez argumentu vypíše velikost pracovního adresáře včetně všech podadresářů # du -sh /home/* (vypíše celkovou velikost domovských adresářů) # du -am /var/log   sort -rn   head -50 (vypíše 50 největších souborů a adresářů v daném adresáři)
<b>touch [&lt;soubor ...   adresar ...&gt;]</b>	vytvoří prázdný soubor; pokud jde o existující soubor nebo adresář, změní čas posledního přístupu a čas poslední změny atributů na aktuální, <b>-a</b> změní jen čas posledního přístupu, <b>-m</b> změní jen čas poslední změny atributů, <b>-d “&lt;RRRR-MM-DD hh:mm:ss&gt;“</b> použije daný čas místo aktuálního, <b>-c</b> nevytvoří žádný soubor

SOUBORY A ADRESÁŘE	
<b>mkdir</b> <adresar ...>	vytvoří adresář, <b>-m</b> <prava> nastaví přístupová práva, <b>-p</b> vytvoří nadřazené adresáře (pokud již neexistují) \$ mkdir -p /adresar1/adresar2/adresar3 (vytvoří adresáře v dané cestě)
<b>mktemp</b>	vytvoří dočasný soubor s jedinečným názvem v adresáři /tmp (vhodné zejména ve skriptech, kde jsou během zpracování vyžadována dočasná data), <b>-d</b> vytvoří dočasný adresář, <b>-p</b> <adresar> určí cílový adresář místo „/tmp“ \$ datasets_temp=\$(mktemp)
<b>mknod</b> <soubor> <typ> [<major> <minor>]	vytvoří speciální soubor daného typu ( <b>b</b> pro blokové zařízení, <b>c</b> pro znakové zařízení, <b>p</b> pro rouru); hlavní (major) a vedlejší (minor) číslo musí být určeno pro blokové či znakové zařízení, <b>-m</b> <prava> nastaví přístupová práva # mknod /dev/sdb b 8 16 (vytvoří speciální soubor pro blokové zařízení „/dev/sdb“ s hlavním číslem 8 a vedlejším číslem 16)
<b>mkfifo</b> <soubor ...>	vytvoří pojmenovanou rouru (FIFO), <b>-m</b> <prava> nastaví přístupová práva
<b>ln</b> <zdroj> [<cil>]	vytvoří pevný odkaz na soubor zadaný prvním argumentem; není-li uveden cíl, vytvoří v pracovním adresáři odkaz na zdroj o stejném názvu, <b>-s</b> symbolický odkaz na soubor či adresář
<b>rm</b> <soubor ...   adresar ...>	odstraní soubor, <b>-d</b> prázdný adresář, <b>-R</b> adresář včetně obsahu, <b>-i</b> žádá o potvrzení, <b>-f</b> provede bez potvrzení a ignoruje neexistující soubory či adresáře, <b>-v</b> podrobný výpis
<b>rmdir</b> <adresar ...>	odstraní prázdný adresář, <b>-p</b> včetně nadřazených adresářů
<b>cp</b> <zdroj> <cil>	kopíruje zdrojový soubor do existujícího cílového adresáře; jedná-li se o dva soubory, přičemž druhý z nich neexistuje, vytvoří se, existuje-li, bude obsah druhého souboru automaticky přepsán, <b>-i</b> žádá o potvrzení, <b>-f</b> provede bez potvrzení, <b>-b</b> vytvoří zálohu cílových souborů, jenž by mohly být případně přepsány, <b>-R</b> kopíruje adresář včetně jeho obsahu (pokud cílový adresář neexistuje, vytvoří ho a zkopíruje do něj pouze obsah zdrojového adresáře), <b>-p</b> zachová atributy zdroje, <b>-u</b> kopíruje jen v případě, že je zdroj novější než cíl nebo cíl neexistuje, <b>-v</b> podrobný výpis, <b>-Z</b> nastaví cílovému souboru bezpečnostní kontext SELinuxu podle výchozí politiky pro dané umístění \$ cp soubor1 ./soubor2 (kopíruje soubor pod jiným názvem do téhož adresáře) # cp -pR /home/tom/{*,.[^.]*} /mnt/extdisk/home/tom (kopíruje obsah domovského adresáře uživatele „tom“ na externí disk, včetně skrytých souborů)
<b>install</b> <zdroj> <cil>	kopíruje zdrojový soubor do existujícího cílového adresáře; jedná-li se o dva soubory, přičemž druhý z nich neexistuje, vytvoří se, existuje-li, bude obsah druhého souboru automaticky přepsán, <b>-b</b> vytvoří zálohu cílových souborů, jenž by mohly být případně přepsány, <b>-m</b> <prava> nastaví přístupová práva, <b>-o</b> <uživatel> nastaví vlastníka souboru, <b>-g</b> <skupina> nastaví skupinu vlastníků souboru, <b>-p</b> zachová atributy zdroje, <b>-d</b> <adresar> vytvoří adresář (včetně nadřazených adresářů), <b>-v</b> podrobný výpis, <b>-Z</b> nastaví cílovému souboru bezpečnostní kontext SELinuxu podle výchozí politiky pro dané umístění # install -m 0755 /bin/chmod.broken /bin/chmod (kopíruje obsah jednoho souboru do druhého a nastaví mu přístupová práva)

SOUBORY A ADRESÁŘE	
<b>rsync</b> [[<uzivatel>@]<hostitel>:]<zdroj ...> [[<uzivatel>@]<hostitel>:]<cil>	<p>kopíruje zdrojový soubor do existujícího cílového adresáře; jedná-li se o dva soubory, přičemž druhý z nich neexistuje, vytvoří se, existuje-li, bude obsah druhého souboru automaticky přepsán; pokud cíl již obsahuje některá data shodná se zdrojem, kopíruje se pouze jejich rozdíl; data lze přenášet i mezi vzdálenými počítači, <b>-r</b> kopíruje adresář včetně jeho obsahu (pokud cílový adresář neexistuje, vytvoří ho), <b>-l</b> zachová symbolické odkazy, <b>-p</b> zachová přístupová práva souboru, <b>-o</b> zachová vlastníka souboru, <b>-g</b> zachová skupinu vlastníků souboru, <b>-t</b> zachová čas poslední změny obsahu souboru, <b>-D</b> zachová speciální soubory, <b>-a</b> zahrnuje volby „-rlptgoD“, <b>-z</b> komprimuje data během přenosu, <b>--delete</b> smaže přebytečná data v cílovém adresáři, která již ve zdroji neexistují, <b>-ignore-errors</b> smaže přebytečná data i v případě I/O chyb, <b>--exclude=&lt;vzor&gt;</b> vyloučí soubory odpovídající danému vzoru, <b>--exclude-from=&lt;soubor&gt;</b> vyloučí soubory odpovídající vzorům obsaženým v daném souboru, <b>--progress</b> vypíše průběh synchronizace, <b>--rsh &lt;prikaz&gt;</b> určí alternativní program pro přenos dat, <b>-v</b> podrobný výpis</p> <pre># rsync -a --delete --exclude ".gvfs" /home /mnt/extdisk (synchronizuje data na externím disku s daty v domovském adresáři) # rsync -avz /var/log/ 192.168.0.20:/logs (kopíruje komprimované logy na vzdálený počítač) \$ rsync --rsh="ssh -l root -p 22022" soubor1 soubor2 192.168.152.47:/tmp (kopíruje soubory na vzdálený počítač pomocí ssh)</pre>
<b>dd</b> <parametr>=<hodnota>	<p>kopíruje data mezi zařízeními, <b>if=&lt;zdroj&gt;</b>, <b>of=&lt;cil&gt;</b>, <b>bs=&lt;velikost_bloku_v_bytech&gt;</b> (implicitně 512 B), <b>count=&lt;pocet_bloku&gt;</b></p> <pre>\$ dd if=/dev/cdrom bs=2048 of=/tmp/image.iso (kopíruje obraz CD do souboru) \$ dd if=/dev/zero of=test bs=5M count=10 (vytvoří prázdný soubor o velikosti 50 MB) \$ for file in {1..50}; do dd if=/dev/zero of=\$file bs=1M count=1; done (vytvoří 50 prázdných souborů o velikosti 1 MB) # dd if=/dev/urandom of=/dev/sdb1 bs=1G (přepíše diskový oddíl náhodnými daty)</pre>
<b>truncate</b> [<soubor ...>]	<p>nastaví, zmenší nebo zvětší velikost souboru, <b>-s</b> určí velikost v B či v jednotkách - „K“, „M“, „G“, „T“ (násobky 1024) či „KB“, „MB“, „GB“, „TB“ (násobky 1000), znak „+“ zvětší velikost, „-“ zmenší velikost, „&lt;“ nastaví maximální velikost, „&gt;“ nastaví minimální velikost, „/“ zaokrouhlí dolů na násobek, „%“ zaokrouhlí nahoru na násobek; pokud soubor neexistuje, vytvoří ho</p> <pre>\$ truncate -s 1M data1 (vytvoří prázdný soubor o velikosti 1 MB) \$ truncate -s "&lt;5M" * (nastaví maximální velikost všech souborů v pracovním adresáři na 5 MB)</pre>
<b>cdrecord</b> <stopa ...>	<p>zapiše data na CD/DVD média, <b>-scanbus</b> najde SCSI adresu CD/DVD rekordéru v systému, <b>dev=&lt;cil&gt;</b> určí SCSI adresu CD/DVD rekordéru, <b>speed=&lt;n&gt;</b> určí rychlost zápisu, <b>-v</b> podrobný výpis</p> <pre>\$ cdrecord -v dev="6,0,0" speed=8 cd.iso (zapiše obsah souboru „cd.iso“ na CD/DVD)</pre>
<b>mv</b> <zdroj> <cil>	<p>přemístí zdrojový soubor či adresář do existujícího cílového adresáře; jedná-li se o dva soubory či dva adresáře, přičemž druhý z nich neexistuje, bude první přejmenován na druhý, pokud však druhý soubor existuje, bude jeho obsah automaticky přepsán, <b>-i</b> žádá o potvrzení, <b>-f</b> provede bez potvrzení, <b>-b</b> vytvoří zálohu cílových souborů, jenž by mohly být případně přepsány, <b>-v</b> podrobný výpis, <b>-Z</b> nastaví cílovému souboru bezpečnostní kontext SELinuxu podle výchozí politiky pro dané umístění</p>
<b>rename</b> <zdroj> <cil> <soubor   adresar>	<p>přejmenuje daný soubor či adresář z původního názvu na cílový podle uvedených vzorů</p> <pre>\$ rename .htm .html *.htm (přejmenuje soubory s příponou „htm“ na „html“) \$ find / -name "honza*"   xargs rename honza jan (přejmenuje soubory a adresáře s názvem „honza*“ na „jan*“)</pre>
<b>basename</b> <soubor   adresar>	<p>vypíše název souboru či adresáře, jehož cesta byla zadána jako argument</p>

SOUBORY A ADRESÁŘE	
<b>dirname</b> <soubor ...   adresar ...>	vypíše cestu k adresáři, v němž se nachází soubor či adresář, jehož cesta byla zadána jako argument <pre># read a; while [[ \$a != "/" ]]; do ls -ld \$a; a=\$(dirname \$a); done</pre> (vypíše práva k adresářům v cestě zadané jako argument)
<b>namei</b> <soubor ...   adresar ...>	prochází danou cestu, rozlišuje symbolické odkazy a uvádí typ souboru pro každou její část („-“ = soubor, „d“ = adresář, „l“ = symbolický odkaz, „p“ = pojmenovaná roura, „b“ = blokové zařízení, „c“ = znakové zařízení), <b>-o</b> vypíše vlastníka a skupinu, <b>-m</b> vypíše přístupová práva, <b>-l</b> vypíše přístupová práva včetně vlastníka a skupiny

SOUBORY A ADRESÁŘE	
<p><b>find</b> [<b>&lt;cesta ...&gt;</b>] [<b>&lt;volba&gt;</b>] [<b>&lt;test&gt;</b>] [<b>&lt;akce&gt;</b>]</p>	<p>hledá soubor či adresář v dané cestě, není-li cesta uvedena, hledá v pracovním adresáři; při použití volby <b>-maxdepth</b> &lt;uroven&gt; prohledává max. do dané úrovně adresářového stromu, <b>-mindepth</b> &lt;uroven&gt; prohledává min. od dané úrovně adresářového stromu; k testování slouží např. <b>-name</b> &lt;vzor&gt; podle názvu odpovídajícímu vzoru, <b>-iname</b> &lt;vzor&gt; podle názvu odpovídajícímu vzoru (nerozlišuje velikost písma), <b>-regex</b> &lt;vzor&gt; podle názvu, jenž odpovídá danému regulárnímu výrazu, <b>-iregex</b> &lt;vzor&gt; podle názvu, jenž odpovídá danému regulárnímu výrazu (nerozlišuje velikost písma), <b>-inum</b> &lt;-n   n   +n&gt; podle i-nodu, <b>-atime</b> / <b>-amin</b> &lt;-n   n   +n&gt; podle posledního přístupu v řádu dní/minut, <b>-ctime</b> / <b>-cmin</b> &lt;-n   n   +n&gt; podle změny atributů v řádu dní/minut, <b>-mtime</b> / <b>-mmin</b> &lt;-n   n   +n&gt; podle změny obsahu v řádu dní/minut, <b>-size</b> &lt;-n   n   +n&gt;&lt;jednotka&gt; podle velikosti („c“ = B, „k“ = kB, „M“ = MB, „G“ = GB), <b>-empty</b> prázdný soubor či adresář, <b>-type</b> podle typu („d“ = adresář, „f“ = soubor, „l“ = symbolický odkaz, „b“ = blokové zařízení, „c“ = znakové zařízení), <b>-context</b> &lt;kontext&gt; podle bezpečnostního kontextu SELinuxu, <b>-user</b> &lt;uživatel   UID&gt; podle vlastníka, <b>-group</b> &lt;skupina   GID&gt; podle skupiny vlastníků, <b>-perm</b> &lt;prava&gt; přesně podle zadaných práv, <b>-perm</b> -&lt;prava&gt; podle nastavených práv, jejichž všechny bity zahrnují alespoň bity zadaných práv, <b>-perm</b> /&lt;prava&gt; podle nastavených práv, jejichž libovolný bit zahrnuje i bit zadaných práv, <b>-nouser</b> hledá soubory neexistujících uživatelů, <b>-nogroup</b> hledá soubory neexistujících skupin; pro akci lze použít např. <b>-exec</b> &lt;prikaz&gt; [&lt;argument ...&gt;] <b>{}</b> <b>{\;</b> <b> \+}</b> provede příkaz (řetězec „{“ je nahrazen cestou k nalezenému souboru, „;“ značí, že daný příkaz je spuštěn zvlášť pro každý odpovídající soubor, zatímco „+“ zpracuje více názvů souborů najednou jako argumenty příkazu; všechny tyto konstrukce by měly být uvozeny nebo potlačeny, aby byly chráněny před expanzí shellem), <b>-ok</b> &lt;prikaz&gt; [&lt;argument ...&gt;] <b>{}</b> <b>\;</b> interaktivní verze volby „-exec“ (vyžaduje potvrzení k provedení operace), <b>-print</b> zobrazí výsledek (implicitně), <b>-print0</b> zobrazí i jména souborů obsahujících mezeru; rozšířené testování umožňují operátory <b>-a</b> = „a“ (oba výrazy jsou pravdivé), <b>-o</b> = „nebo“ (alespoň jeden výraz je pravdivý), <b>!</b> = negace (výraz je nepravdivý); bez argumentu vypíše obsah pracovního adresáře rekurzivně</p> <pre># find /home -name "*.jpg" -o -name "*.png" (hledá v domovském adresáři soubory s příponou „.jpg“ nebo „.png“) # find . -type f -o -type d   wc -l (spočítá v pracovním adresáři všechny soubory a adresáře) # find ! -type f (vyhledá v pracovním adresáři vše kromě běžných souborů) # find /tmp -mmin -120 (vyhledá všechny soubory, které byly změněny za poslední 2 h) # find / -user tom -exec rm -Rf '{}' \; (smaže všechny soubory a adresáře uživatele „tom“) # find . -maxdepth 2 -type l (hledá symbolické odkazy do druhé úrovně adresářového stromu) # find / -nouser -o -nogroup (hledá soubory neexistujících uživatelů a skupin) # find / -perm -4000 (vyhledá všechny soubory s nastaveným SUID bitem) # find / -perm -444 -perm /222 ! -perm /111 (vyhledá soubory, které jsou čitelné pro všechny, mají nastaven aspoň 1 bit pro zápis, ale nejsou spustitelné pro nikoho) # find / -context "*tmp_t*" (hledá soubory podle daného bezpečnostního kontextu SELinuxu) # find /var/log -type f -size +100M -exec du -h '{}' \+   sort -rh   head -10 (vyhledá a sestupně seřadí 10 souborů větších než 100 MB) # find /etc/rc.d -type f \( -perm -g=w -o -perm -o=w \) (vyhledá soubory s právy k zápisu pro skupinu nebo ostatní, kromě symbolických odkazů) # find / -type d \( -perm -o=w -a ! -perm -o=t \) -exec ls -ld '{}' \; (vyhledá adresáře s právem zápisu pro ostatní bez nastaveného sticky bitu) # find / -type f -regex ".*\.(sh pl py)" -perm -o=x (vyhledá skripty s právy ke spuštění pro ostatní)</pre>

SOUBORY A ADRESÁŘE	
<b>which</b> <prikaz ...>	vypíše absolutní cestu daného příkazu nebo skriptu (jeho aliasu) v \$PATH, <b>-a</b> zobrazí všechny výskyty v \$PATH
<b>whereis</b> <prikaz ...>	vypíše absolutní cestu k binárnímu, zdrojovému a manuálovému souboru daného příkazu nebo skriptu v \$PATH
<b>type</b> <prikaz ...>	vypíše, zda se jedná o program či vestavěný příkaz nebo klíčové slovo shellu, u programu zobrazí absolutní cestu k binárnímu souboru, <b>-a</b> vypíše všechny výskyty včetně aliasů a funkcí
<b>apropos</b>   <b>man -k</b> <klicove_slovo ...>	hledá příkazy obsahující v názvu či popisu daný řetězec
<b>whatis</b>   <b>man -f</b> <klicove_slovo ...>	hledá příkazy, jejichž název odpovídá danému řetězci
<b>man</b>   <b>info</b> <prikaz>	zobrazí manuálovou stránku k danému příkazu, <b>q</b> ukončí program
<b>help</b> [<vzor>]	vypíše nápovědu k vestavěným příkazům a klíčovým slovům shellu odpovídajícím vzoru; bez argumentu vypíše všechny vestavěné příkazy a klíčová slova shellu
<b>ldd</b> <program ...>	vypíše sdílené knihovny požadované daným programem
<b>ls</b> [<soubor ...>]	vypíše informace o daném otevřeném souboru - např. jeho název, program, který soubor používá, PID, uživatele, souborový popisovač, typ souboru, hlavní (major) a vedlejší (minor) číslo zařízení, velikost souboru v B a číslo i-uzlu, <b>+D</b> <adresar> vypíše otevřené soubory v daném adresáři rekurzivně, <b>-l</b> vypíše UID, <b>-p</b> <PID> soubory otevřené daným procesem, <b>-u</b> <uzivatel> soubory otevřené procesy vlastněnými daným uživatelem, <b>-i</b> vypíše síťové soubory, <b>-n</b> nepřeloží jména počítačů pomocí DNS, <b>-P</b> nepřeloží čísla portů na názvy portů, <b>-r</b> <n> aktualizuje výpis každých <i>n</i> sekund; bez argumentu se zobrazí všechny otevřené soubory patřící všem aktivním procesům; otevřený soubor může být běžný soubor, adresář, blokový speciální soubor, znakový speciální soubor, knihovna nebo síťový soubor (internetový socket, NFS soubor či socket UNIX domény) <pre>\$ ls -l /dev/cdrom</pre> (vypíše podrobné informace o procesech používajících CD/DVD mechaniku) <pre>\$ ls -l /dev/snd/*</pre> (vypíše podrobné informace o procesech používajících zvukovou kartu) <pre># ls -l +D /opt/app   awk 'NR &gt; 1 {print \$9}' &gt; /tmp/otevrene_soubory.txt</pre> (uloží seznam otevřených souborů v adresáři „/opt/app“ do „/tmp/otevrene_soubory.txt“) <pre>\$ ls -p</pre> (vypíše otevřené soubory z aktuálního shellu) <pre># ls -i -n -P</pre> (vypíše všechny otevřené porty včetně typů protokolů, IP adres hostitelů, uživatelů a procesů používajících daný port)

Obsah souboru	
<b>file</b> <soubor ...>	zjistí typ datového formátu běžného souboru (ASCII, PDF, HTML atd.), <b>-s</b> čte blokové nebo znakové speciální soubory, <b>-z</b> čte obsah komprimovaného souboru <pre>\$ file *   grep ASCII</pre> (vypíše soubory v pracovním adresáři obsahující prostý text)
<b>cat</b> [<soubor ...>]	vypíše obsah textového souboru, <b>-n</b> čísluje všechny řádky, <b>-b</b> čísluje jen řádky obsahující text, <b>-s</b> potlačí opakující se prázdné řádky; není-li soubor uveden, čte ze STDIN <pre>\$ cat a b c &gt; abc</pre> (zapíše obsah tří souborů do souboru „abc“)
<b>tac</b> [<soubor ...>]	vypíše obsah textového souboru v opačném pořadí; není-li soubor uveden, čte ze STDIN
<b>more</b> <soubor ...>	vypíše obsah většího textového souboru po stránkách, <b>mezera</b> posune o stránku vpřed, <b>b</b> o stránku vzad, <b>/&lt;vyraz&gt;</b> hledá v textu daný výraz směrem vpřed, <b>n</b> pokračuje v dalším hledání, <b>q</b> ukončí program
<b>less</b> <soubor ...>	vypíše obsah většího textového souboru po stránkách, <b>mezera</b> / <b>page down</b> posune o stránku vpřed, <b>b</b> / <b>page up</b> o stránku vzad, <b>&lt;</b> na začátek souboru, <b>&gt;</b> na konec souboru, <b>/&lt;vyraz&gt;</b> hledá v textu daný výraz směrem vpřed, <b>n/N</b> pokračuje v dalším hledání / v hledání opačným směrem, <b>-N</b> čísluje řádky, <b>:n</b> vypíše obsah dalšího souboru, <b>:p</b> vypíše obsah předchozího souboru, <b>q</b> ukončí program
<b>head</b> [<soubor ...>]	<b>[-]&lt;n&gt;</b> vypíše prvních <i>n</i> řádků souboru (implicitně 10), je-li před číslem uveden znak '-', vypíše všechny kromě posledních <i>n</i> řádků; není-li soubor uveden, čte ze STDIN
<b>tail</b> [<soubor ...>]	<b>[+&lt;n&gt;</b> ] vypíše posledních <i>n</i> řádků souboru (implicitně 10), je-li před číslem uveden znak '+', vypíše řádky začínající řádkem <i>n</i> , <b>-f</b> aktualizuje výpis o nově příchozí data; není-li soubor uveden, čte ze STDIN <pre>\$ head -25 access.log   tail -1</pre> (vypíše 25. řádek souboru)

Obsah souboru	
<b>wc</b> [<soubor ...>]	vypíše počet řádků, slov a bytů v textovém souboru, včetně bílých míst, <b>-l</b> počítá řádky, <b>-w</b> slova, <b>-m</b> znaky, <b>-c</b> byty; není-li soubor uveden, čte ze STDIN
<b>nl</b> [<soubor ...>]	čísluje řádky souboru, <b>-v</b> <n> určí číslo, kterým má číslování začít (implicitně 1), <b>-i</b> <n> určí přírůstek, o který se číslo následujícího řádku zvýší (implicitně 1), <b>-b</b> <volba> čísluje řádky odpovídající dané volbě ( <b>a</b> všechny řádky, <b>t</b> neprázdné řádky (implicitně), <b>n</b> žádné řádky, <b>p</b> <vyraz> řádky obsahující daný regulární výraz), <b>-s</b> <retezec> určí oddělovač mezi číslem řádku a jeho obsahem (implicitně tabulátor), <b>-w</b> <n> určí počet znaků (míst) pro zarovnání jednotlivých číslic vpravo (implicitně 6); není-li soubor uveden, čte ze STDIN \$ nl style.css   more (čísluje řádky souboru, jehož obsah zobrazí po stránkách) \$ nl -s " " -b p^# script.sh (čísluje řádky souboru začínající znakem „#“)
<b>sort</b> [<soubor ...>]	vzestupně seřadí řádky textového souboru (podle abecedy), <b>-n</b> podle číselných hodnot, <b>-h</b> podle číselných hodnot uvedených v čitelném formátu, <b>-k</b> <n> podle sloupce, <b>-u</b> ignoruje duplicitní řádky, <b>-r</b> v obráceném pořadí; není-li soubor uveden, čte ze STDIN \$ sort soubor.txt   uniq (setřídí a vypíše jedinečné řádky souboru)
<b>uniq</b> [<soubor>]	vypíše obsah textového souboru, následují-li po sobě identické řádky, zobrazí vždy jen jeden z nich, <b>-d</b> vypíše duplicitní, po sobě jdoucí řádky (pro výpis všech duplicitních řádků je třeba obsah souboru setřídít), <b>-u</b> jedinečné řádky, <b>-c</b> u každého řádku vypíše počet výskytů totožných, po sobě jdoucích řádků, <b>-i</b> ignoruje velikost písmen; není-li soubor uveden, čte ze STDIN
<b>grep</b> <vzor> [<soubor ...   adresar ...>]	vyhledá v textovém souboru řetězce obsahující daný vzor za použití základních regulárních výrazů a vypíše odpovídající řádek, <b>-v</b> vypíše řádky, které daný vzor neobsahují, <b>-n</b> vypíše číslo řádku, <b>-c</b> vypíše počet řádků obsahujících daný vzor, <b>-l</b> vypíše jména souborů obsahujících daný vzor, <b>-L</b> vypíše jména souborů, které daný vzor neobsahují, <b>-h</b> potlačí výpis názvu souborů na výstupu, <b>-i</b> ignoruje velikost písmen, <b>-f</b> <soubor> čte vzory z daného souboru (na každém řádku jeden), <b>-o</b> vypíše jen tu část řádku odpovídající vzoru, <b>-A</b> <n> vypíše dalších <i>n</i> řádků po řádku odpovídajícím shodě, <b>-B</b> <n> vypíše dalších <i>n</i> řádků před řádkem odpovídajícím shodě, <b>-C</b> <n> vypíše dalších <i>n</i> řádků před i po řádku odpovídajícím shodě, <b>-E</b> interpretuje vzory jako rozšířené regulární výrazy (obdoba příkazu „egrep“), <b>-F</b> interpretuje vzory jako pevné řetězce (obdoba příkazu „fgrep“), <b>-R</b> rekurzivně, <b>-w</b> vyhledá celá slova, jimž vzor vyhovuje, <b>--color</b> barevně zvýrazní odpovídající řetězce; není-li soubor uveden, čte ze STDIN # grep -i linux /etc/httpd/logs/access_log (vypíše řádky souboru obsahující výraz „linux“ v kombinaci malých i velkých písmen) \$ ls -l   grep '^d' (vypíše podadresáře v pracovním adresáři) \$ ps -ef   grep -w "httpd" (vypíše procesy webového serveru) \$ grep '^P.*r\$' notes (vypíše řádky souboru začínající na „P“ a končící na „r“) \$ grep -Ei 'red hat centos fedora' /etc/*release (vypíše, zda systém odpovídá jedné z uvedených distribucí Linuxu) \$ grep -R "header" /var/www/html (vypíše všechny řádky všech souborů v daném adresáři obsahující výraz „header“)
<b>egrep</b>   <b>grep -E</b> <vzor> [<soubor ...   adresar ...>]	vyhledá v textovém souboru řetězce obsahující daný vzor za použití rozšířených regulárních výrazů a vypíše odpovídající řádek; není-li soubor uveden, čte ze STDIN # egrep -v '^[#;] ^\$' *conf* (vypíše všechny řádky konfiguračních souborů, které nejsou zakomentované nebo prázdné) \$ egrep '^[^#]*authpriv\.\' /etc/rsyslog.conf (vypíše řádky souboru začínající výrazem „authpriv.“) \$ egrep -o '^[^:]+\' /etc/passwd (vypíše všechny uživatele v systému)
<b>fgrep</b>   <b>grep -F</b> <vzor> [<soubor ...   adresar ...>]	vyhledá v textovém souboru řetězce obsahující daný vzor bez použití regulárních výrazů a vypíše odpovídající řádek; není-li soubor uveden, čte ze STDIN \$ fgrep '/' style.css (vypíše řádky souboru obsahující komentář)

Obsah souboru	
<p><b>sed</b> [<b>&lt;adresa&gt;</b>] <b>&lt;prikaz&gt;</b> [<b>&lt;soubor ...&gt;</b>]</p>	<p>filtruje a edituje text neinteraktivním způsobem (využití zejména ve skriptech), <b>-E</b> umožní použití rozšířených regulárních výrazů, <b>-f &lt;soubor&gt;</b> provede skript obsažený v daném souboru, <b>-i</b> provede změny přímo v souboru (jinak je nutné změny uložit použitím přesměrování), <b>-n</b> nevyíše výstup na STDOUT; adresa představuje buď pořadové číslo řádku od začátku vstupu (např. 1,10 = rozmezí řádků), znak „\$“ (poslední řádek vstupu) či vzorek (regulární výraz) ohraničený z obou stran oddělovačem - standardně „/“; příkaz <b>s</b> nahradí vzorek následujícím výrazem ohraničeným oddělovači (jen první vyhovující vzorek), <b>n</b> právě ntý výskyt vzorku, <b>g</b> nahradí vzorek globálně, u každého výskytu, <b>a</b> přidá výraz pod řádek se zadaným vzorem, <b>&lt;n&gt;p</b> vypíše daný řádek souboru (s volbou „-n“), <b>d</b> smaže celý řádek obsahující daný výraz, <b>!</b>&lt;prikaz&gt; provede příkaz pro všechny adresy, kromě adres uvedených; není-li soubor uveden, čte ze STDIN</p> <pre>\$ echo "11 x, 22 x, 33 x"   sed 's/x/y/2' (nahradí pouze druhý výskyt řetězce „x“ řetězcem „y“) \$ sed -i 's/retezec1/retezec2/g' soubor (nahradí první řetězec druhým v celém souboru) # sed -i '/shared/s/ro/rw/' /etc/fstab (nahradí na řádku s prvním výskytem slova „shared“ řetězec „ro“ řetězcem „rw“) \$ sed 's/ \\/\t/g' test1 &gt; test2 (nahradí v souboru různý počet mezer jedním tabulátorem) \$ echo "Log retention:" \$(sed -n '3p;6p' /etc/logrotate.conf) (vypíše 3. a 6. řádek souboru) \$ sed '/dev/!d' soubor (vypíše řádky obsahující výraz „dev“) \$ sed -i '1,10d' soubor (smaže prvních 10 řádků souboru) \$ sed -i '5,\$d' soubor (smaže vše od 5. řádku až po konec souboru) \$ sed '1,/START/d' soubor (smaže vše do slova „START“ včetně) \$ sed '1,/STOP/!d' soubor (smaže vše od slova „STOP“) \$ sed -n 'p;n' soubor (vypíše liché řádky souboru) \$ sed -n 'n;p' soubor (vypíše sudé řádky souboru) \$ sed '/^#/d; /^\$/d' soubor1 &gt; soubor2   \$ sed -E '/^(# \$)/d' soubor1 &gt; soubor2 (odstraní v souboru komentáře a prázdné řádky) # sed -i '\$s/^/from="\$\$(awk -F " " '{print \$9 "," \$11}' hosts.csv)"/' / .ssh/authorized_keys (vloží výstup příkazu před text na prvním řádku) # sed -i '\$s/\$/ !!Ansible user!!/' .ssh/authorized_keys (vloží řetězec za text na posledním řádku) # sed -i '/auth[[:blank:]]+required[[:blank:]]\+pam_env.so/a auth\trequired\tpam_tally2.so deny=5' /etc/pam.d/system-auth (přidá výraz pod řádek se zadaným vzorem) \$ sed '/^[^#]*\([sha[25]\ md5\)/!d' /etc/pam.d/system-auth /etc/pam.d/common-password 2&gt; /dev/null   egrep -o "sha[1256]{3} md5"    sed '/^[^#]*ENCRYPT_METHOD/!d' /etc/login.defs (zobrazí šifrovací algoritmus pro nově vytvořená hesla)</pre>
<p><b>awk</b> '<b>&lt;program&gt;</b>' [<b>&lt;soubor ...&gt;</b>]</p>	<p>filtruje text neinteraktivním způsobem (využití zejména ve skriptech); program je série pravidel obsahující vzorek, akci či obojí, akce je uzavřena do „{}“; <b>-f &lt;soubor&gt;</b> čte program z daného souboru, <b>-F &lt;retezec&gt;</b> určí oddělovač položek vstupu, <b>-v &lt;promenna&gt;=&lt;hodnota&gt;</b> nastaví danou proměnnou v rámci programu; vestavěná proměnná <b>NR</b> představuje počet záznamů (řádků) oddělených novým řádkem, <b>NF</b> představuje počet položek na řádku oddělených oddělovačem (implicitně bílým místem), na jednotlivé položky se odkazuje \$1, \$2 atd., \$0 odkazuje na celý záznam, <b>RS</b> definuje oddělovač záznamů, <b>FS</b> definuje oddělovač položek, <b>ORS</b> definuje výstupní oddělovač záznamů, <b>OFS</b> definuje výstupní oddělovač položek; pro porovnání řetězců slouží operátory == je rovno, != není rovno, &lt; méně než, &gt; více než, &lt;= méně než nebo rovno, &gt;= více než nebo rovno; regulární výrazy jsou uzavřeny mezi „/“/“, pro porovnání slouží ~ (vzor odpovídá výrazu), !~ (vzor neodpovídá výrazu);    logické NEBO, &amp;&amp; logické A; není-li soubor uveden, čte ze STDIN</p> <pre>\$ awk '/nameserver/ {print}' /etc/resolv.conf (vypíše všechny řádky obsahující „nameserver“) \$ awk -F ":" '{print \$1, \$7}' /etc/passwd (vypíše všechny uživatele a jejich přihlašovací shell) \$ awk '{print \$1 &gt; "soubor1"; print \$2 &gt; "soubor2"}' test (zkopíruje první a druhé slovo souboru do zadaných souborů) \$ awk 'length &gt; 10' seznam.txt (vypíše řádky delší než deset znaků) \$ awk 'NF &lt; 5' seznam.txt (vypíše řádky kratší než pět slov) \$ awk 'NR == 5' error.log (vypíše 5. řádek souboru) \$ awk 'NR == 5, NR == 10' album (vypíše 5. až 10. řádek souboru) \$ awk '{print NR, "-&gt;", NF}' seznam.txt (vypíše počet slov u každého řádku souboru) \$ ls -l   grep "^-"   awk '{sum += \$5} END {print sum/1024/1024}' (vypíše velikost všech souborů v pracovním adresáři v MB) # ne=\$(awk -F ":" '{if (length(\$2) &gt; 2 &amp;&amp; (\$5 == ""    \$5 &gt;= 99999)) print \$1}' /etc/shadow); [[ -z "\$ne" ]] &amp;&amp; echo "NONE"    echo "\$ne"   tr " " "\n" (vypíše uživatele s neomezenou dobou platnosti hesla) \$ sed '/^[^:]/!d' /etc/inittab   awk -F "#" '{print \$1}'   awk -F ":" 'if ((\$4 !~ /\^\/ &amp;&amp; \$4 !~ /\^\$/ )    (\$4 ~ /[\+.\?\/] &amp;&amp; \$4 !~ /\^\$/)) print \$4' (vyhledá v souboru příkazy, které nejsou uvedeny absolutní cestou)</pre>

Obsah souboru	
<b>tr</b> <retezec1> [<retezec2>]	nahradí jednotlivé znaky z prvního řetězce novými z druhého (původní soubor nepřepíše), <b>-d</b> smaže znaky, <b>-s</b> opakující se znaky nahradí jedním výskytem <pre>\$ echo "FIBRE CHANNEL STATISTICS REPORT"   tr 'A-Z' 'a-z'</pre> <p>(převede velká písmena na malá písmena)</p> <pre>\$ cat soubor1   tr 'a-z' 'A-Z' &gt; soubor2</pre> <p>(převede v souboru malá písmena na velká a obsah přesměruje do nového souboru)</p> <pre>\$ cat text1   tr ' ' '\t' &gt; text2</pre> <p>(nahradí v souboru každou mezeru tabulátorem)</p> <pre>\$ awk -F ":" '{print \$1}' /etc/passwd   tr '\n' ' '</pre> <p>(nahradí nový řádek mezerou)</p> <pre>\$ tr -d ';' &lt; text1.txt &gt; text2.txt</pre> <p>(smaže daný znak v souboru a obsah přesměruje do nového souboru)</p>
<b>rev</b> [<soubor ...>]	obrátlí pořadí znaků na každém řádku souboru (původní soubor nepřepíše); není-li soubor uveden, čte ze STDIN <pre>\$ echo madam   rev</pre>
<b>tee</b> [<soubor ...>]	čte ze STDIN a zapisuje na STDOUT a případně do daného souboru, <b>-a</b> výstup příkazu přidá na konec daného souboru; pokud soubor neexistuje, vytvoří ho <pre>\$ ls   tee /tmp/test   wc -l</pre> <p>(výstup příkazu „ls“ zapíše do souboru „/tmp/test“, na STDOUT zobrazí pouze počet řádků)</p> <pre>\$ who   sort   tee -a log1 log2</pre> <p>(setříděný výstup příkazu „who“ zapíše jak na STDOUT, tak do souborů „log1“ a „log2“)</p>
<b>cat</b> > <soubor>	vloží obsah STDIN do daného souboru <pre>\$ cat &gt; /dev/pts/2 &lt;- '</pre> <p>&lt;text&gt; &lt;-'</p> <p><b>Ctrl+d</b></p> <p>(přesměruje uvedený text na daný terminál)</p>
>   <b>cat /dev/null</b> > <soubor>	smaže obsah souboru

Obsah souboru	
<p><b>vi   vim</b> &lt;soubor&gt;  <b>v</b>  <b>Shift+v</b>  <b>Ctrl+v</b>  <b>Esc</b>                  :&lt;n&gt;                  :q                  :q!                  :w / :w &lt;soubor&gt;                  :wq / :x / ZZ                  :w! / :wq!                  :set number / nonumber                  :set ignorecase / noignorecase                  :set all</p> <p><b>pohyb kurzoru:</b>                  vlevo:  <b>h / Šipka vlevo</b>  <b>b</b>  <b>O</b>                  vpravo:  <b>l / Šipka vpravo</b>  <b>w</b>  <b>\$</b>                  nahoru:  <b>k / Šipka nahoru</b>  <b>H</b>  <b>1G</b>                  dolů:  <b>j / Šipka dolů</b>  <b>L</b>  <b>G</b></p> <p><b>přidání textu:</b>  <b>i</b>  <b>a</b>  <b>I</b>  <b>A</b>  <b>o</b>  <b>O</b></p> <p><b>mazání / vyjmutí textu:</b>  <b>x</b>  <b>db</b>  <b>dw</b>  <b>d0</b>  <b>d\$</b>  <b>dd / &lt;n&gt;dd</b>  <b>d1G</b>  <b>dG</b>                  :1,\$d</p> <p><b>úprava textu:</b>  <b>yw / &lt;n&gt;yw</b>  <b>yy / &lt;n&gt;yy</b>  <b>P / p</b>  <b>r</b>  <b>R</b>                  :[&lt;adresa&gt;]s/&lt;hledany_vyraz&gt;/&lt;nahrazujici_vyraz&gt;(/&lt;volba&gt;)</p> <p><b>u</b>  <b>U</b>                  .</p> <p><b>hledání v textu:</b>                  /&lt;vyraz&gt;                  ?&lt;vyraz&gt;                  n / N</p>	<p>spustí textový editor „vi“/„vim“ a vytvoří či otevře soubor                  spustí/ukončí vizuální znakový režim                  spustí/ukončí vizuální řádkový režim                  spustí/ukončí vizuální blokový režim                  přepne z režimu vkládání do příkazového režimu                  přemístí kurzor na začátek daného řádku                  ukončí editor (pokud soubor nebyl změněn)                  ukončí editor bez uložení případných změn v souboru                  uloží změny v souboru / uloží soubor jako nový                  uloží změny a ukončí editor                  uloží změny / uloží změny a ukončí editor u souboru s právy pouze ke čtení                  očísluje řádky / odstraní číslování řádků                  nerozlišuje/rozlišuje velikost písmen                  vypíše nastavení všech voleb</p> <p>jeden znak vlevo                  na začátek předchozího slova                  na začátek řádku</p> <p>jeden znak vpravo                  na začátek následujícího slova                  na konec řádku</p> <p>na předchozí řádek                  na první řádek obrazovky                  na první řádek souboru</p> <p>na následující řádek                  na poslední řádek obrazovky                  na poslední řádek souboru</p> <p>v místě kurzoru                  za kurzor                  na začátek řádku                  na konec řádku                  na začátek nového řádku pod aktuálním řádkem                  na začátek nového řádku nad aktuálním řádkem</p> <p>jeden znak v místě kurzoru                  znaky před kurzorem až do začátku slova                  znak v místě kurzoru až do konce slova                  znaky před kurzorem až do začátku řádku                  znak v místě kurzoru až do konce řádku                  celý řádek v místě kurzoru / následujících n řádků                  celý řádek v místě kurzoru až do začátku souboru                  celý řádek v místě kurzoru až do konce souboru                  obsah celého souboru</p> <p>kopíruje znaky v místě kurzoru až do konce slova / následujících n slov                  kopíruje současný řádek / následujících n řádků                  vloží vyjmutý či zkopírovaný text před/za kurzor nebo nad/pod současný řádek (v případě vyjmutí/kopie celých řádků)                  nahradí znak v místě kurzoru                  nahradí znaky v místě kurzoru                  nahradí hledaný výraz výrazem novým; adresa 1,\$ označuje rozsah celého dokumentu, není-li adresa uvedena, změna se projeví pouze na aktuálním řádku, volba g provede náhradu všech výskytů hledaného řetězce, volba c vyžaduje potvrzení k provedení operace                  :1,\$s/sto/100/g                  (nahradí všechny výskyty slova "sto" číslem "100" v celém souboru)                  vrátí zpět o poslední provedenou změnu                  vrátí zpět o všechny změny provedené na současném řádku                  opakuje poslední příkaz</p> <p>hledá v textu daný výraz směrem vpřed                  hledá v textu daný výraz směrem vzad                  pokračuje v dalším hledání / v hledání opačným směrem</p>
<p><b>column</b> [&lt;soubor ...&gt;]</p>	<p>formátuje řádky souboru do sloupců, -s &lt;oddělovač&gt; určí oddělovač vstupních položek (implicitně bílá místa), -t vytvoří tabulku; není-li soubor uveden, čte ze STDIN                  \$ mount   column -t                  (formátuje výstup příkazu do tabulky)</p>
<p><b>cut</b> [&lt;soubor ...&gt;]</p>	<p>-c &lt;m-n&gt; vyřízne v souboru na každém řádku dané rozpětí znaků (od začátku řádku), -f &lt;m-n&gt; vyřízne dané rozpětí sloupců (polí) oddělených tabulátorem, -d &lt;znak&gt; určí oddělovač polí místo tabulátoru; položky seznamu se oddělují čárkou; není-li soubor uveden, čte ze STDIN                  \$ cut -f 2,4 -d ' ' data                  (vyřízne ze souboru sloupce 2 a 4 a oddělí je mezerou)                  # cut -d : -f 1 &lt;/etc/passwd   sort                  (vypíše všechny uživatele systému v abecedním pořadí)</p>
<p><b>paste</b> [&lt;soubor1&gt; &lt;soubor2&gt;]</p>	<p>spojí obsah souborů vedle sebe (1 sloupec = 1 soubor), -d &lt;retezec&gt; určí oddělovač polí místo tabulátoru, -s transpozice řádků a sloupců (1 řádek = 1 soubor); není-li soubor uveden, čte ze STDIN                  \$ paste -s soubor1 soubor2 &gt; soubor3                  (spojí obsah dvou souborů do jednoho - 1 řádek = 1 soubor)</p>

<b>Obsah souboru</b>	
<b>split</b> [<soubor>]	rozdělí soubor na několik částí o stejném počtu řádků (implicitně 1000), <b>-l</b> <n> určí počet řádků, <b>-b</b> <n> rozdělí soubor podle daného počtu bytů; není-li soubor uveden, čte ze STDIN
<b>join</b> <soubor1> [<soubor2>]	spojí řádky dvou setříděných textových souborů podle prvního, identického pole odděleného bílým místem, <b>-i</b> ignoruje velikost písmen; není-li soubor uveden, čte ze STDIN
<b>diff</b> <soubor1> <soubor2>   <adresar1> <adresar2>	porovná obsah dvou textových souborů nebo adresářů a vypíše rozdíl, <b>-i</b> ignoruje velikost písmen, <b>-q</b> hlásí, když se soubory liší, <b>-s</b> hlásí, když jsou soubory stejné, <b>-r</b> rekurzivně, <b>-u</b> zobrazí výpis v jednotném tvaru \$ diff <(echo "Vítejte na prompt.cz") <(cat /etc/motd)> (porovná výstup obou příkazů)
<b>cmp</b> <soubor1> <soubor2>	porovná obsah dvou souborů libovolného typu a vypíše první odlišné číslo bytu a řádku, <b>-l</b> vypíše všechny rozdíly byte po byte
<b>sum</b> [<soubor ...>]	vypíše kontrolní součet a počet bloků souboru; není-li soubor uveden, čte ze STDIN
<b>cksum</b> [<soubor ...>]	vypíše CRC kontrolní součet a počet bytů souboru; není-li soubor uveden, čte ze STDIN
<b>sha1sum</b>   <b>sha512sum</b>   <b>md5sum</b> [<soubor ...>]	vypíše nebo ověří SHA1 (160-bit), SHA512 (512-bit) či MD5 (128-bit) kontrolní součet souboru, <b>-c</b> <soubor> porovná kontrolní součet uložený v daném souboru s původním souborem; není-li soubor uveden, čte ze STDIN \$ sha512sum ./F-7-x86_64-DVD.iso (vypíše kontrolní součet souboru) \$ md5sum page.txt > page.md5; md5sum -c page.md5 (uloží kontrolní součet souboru „page.txt“ do „page.md5“ a poté ho porovná s původním souborem)
<b>ksvalidator</b> <soubor ...>	ověří syntaxi souboru kickstart (automaticky vytvořeného instalačním programem Anaconda a uloženého jako /root/anaconda-ks.cfg)
<b>enca</b> <soubor ...>	zjistí typ kódování textového souboru, <b>-L</b> <jazyk> určí jazyk vstupního souboru \$ enca -L cs input.srt (zjistí typ kódování textového souboru, pokud program automaticky nerozpoznal použitý jazyk)
<b>convmv</b> <soubor ...   adresar ...>	změní typ kódování jména souboru/adresáře, <b>-f</b> <zdrojovy_kod> určí původní kód, <b>-t</b> <cilovy_kod> určí nový kód, <b>-i</b> interaktivní režim, <b>-r</b> rekurzivně, <b>--list</b> vypíše všechna dostupná kódování, <b>--notest</b> provede operaci \$ convmv --notest -f cp1250 -t UTF-8 * (změní typ kódování jmen obsahu pracovního adresáře z „cp1250“ na „UTF-8“)
<b>iconv</b> [<soubor ...>]	změní typ kódování obsahu textového souboru, <b>-f</b> <zdrojovy_kod> určí původní kód, <b>-t</b> <cilovy_kod> určí nový kód, <b>-l</b> vypíše všechna dostupná kódování; není-li soubor uveden, překóduje STDIN na STDOUT \$ iconv -f WINDOWS-1250 -t UTF-8 < input.srt > output.srt (změní typ kódování obsahu textového souboru z „WINDOWS-1250“ na „UTF-8“)
<b>recode</b> [<zdrojovy_kod>].<cilovy_kod> [<soubor ...>]	změní typ kódování obsahu textového souboru, <b>-l</b> vypíše známé typy kódování, <b>-f</b> potlačí chybový výstup, <b>-v</b> podrobný výpis; není-li soubor uveden, překóduje STDIN na STDOUT \$ echo 'Výpis jmen všech souborů'   recode -f UTF-8..flatt (vypíše text bez diakritiky) \$ recode .HTML < page.txt > page.html (změní typ kódování obsahu souboru „page.txt“ na „html“ a zapíše ho do „page.html“) \$ find . -name "*.txt" -exec recode cp1250..UTF-8 '{}' \; (změní typ kódování obsahu textových souborů z „cp1250“ na „UTF-8“)
<b>od</b> [<soubor ...>]	[ <b>-o</b> ] vypíše obsah binárního souboru oktálově, <b>-d</b> decimálně, <b>-x</b> hexadecimálně, <b>-j</b> <byte> vypíše data od daného bytu (offsetu), <b>-w</b> <pocet_bytu> zobrazí na řádku daný počet bytů; první sloupec výpisu označuje tzv. offset - pořadí prvního bytu daného řádku od začátku souboru; není-li soubor uveden, čte ze STDIN \$ od -w8 /usr/bin/who \$ od -j 0474360 /usr/bin/find
<b>strings</b> <soubor ...>	vypíše textové řetězce obsažené zejména uvnitř binárního souboru o minimální délce čtyř znaků # strings \$(which vsftpd)   egrep "libwrap hosts" (zjistí, zda se program vztahuje k tcp wrappers)
<b>msgfmt</b> <soubor>.po	převede textový soubor do binárního formátu, <b>-o</b> <soubor>.mo určí výstupní soubor \$ msgfmt -o woocommerce-cs_CZ.mo woocommerce-cs_CZ.po
<b>convert</b> <zdroj> <cil>	změní formát souboru, případně i jeho velikost \$ convert image.tif image.jpg (změní formát souboru „tif“ na „jpg“) \$ IFS=', '; cmds="hostname -s,date,ifconfig -a"; for cmd in \$cmds; do { echo "\${USER}@\${HOSTNAME} \${PWD/#\$HOME/~}" "\${cmd}";}; eval "\${cmd}"; done > info.txt && convert -background black -fill white info.txt info.png (vytvoří printscreen terminálu se zadanými příkazy)
<b>lame</b> <soubor>.wav <soubor>.mp3	změní formát audio souboru z „wav“ na „mp3“ či obráceně, <b>-b</b> <bitrate> určí minimální bitrate, <b>-v</b> variabilní bitrate, <b>--verbose</b> podrobný výpis
<b>aplay</b> <soubor ...>	přehraje daný audio soubor
<b>arecord</b> <soubor>.wav	vytvoří „wav“ audio soubor

Archivace a komprimace	
<b>tar</b> [-f <archiv>] [<soubor ...   adresar ...>]	<p>archivuje či obnovuje data (původní zdroj zachová), <b>-c</b> vytvoří archiv, <b>-f</b> &lt;soubor&gt; určí název archivu, <b>-v</b> podrobný výpis, <b>--remove-files</b> smaže původní zdroj, <b>--acls</b> zachová přístupová práva ACL, <b>--selinux</b> zachová nastavení SELinuxu, <b>-M</b> vícesvazkový archiv (data vyžadují více než 1 medium), <b>-V</b> &lt;jmeno&gt; určí název svazku, <b>-X</b> &lt;soubor&gt; vyloučí soubory odpovídající vzorům uvedeným v daném souboru, <b>-d</b> porovná data v archivu s daty na disku, <b>-t</b> vypíše obsah archivu (i komprimovaného), <b>-r</b> připojí soubory na konec archivu, <b>-x</b> rozbalí archiv, <b>-C</b> přejde do adresáře (určí cílový adresář pro rozbalení archivu), <b>-p</b> zachová původního vlastníka a přístupová práva během obnovy dat, <b>-Z</b> (de)komprimuje nástrojem „compress“, <b>-z</b> (de)komprimuje nástrojem „gzip“, <b>-j</b> (de)komprimuje nástrojem „bzip2“, <b>-J</b> (de)komprimuje nástrojem „xz“, <b>-w</b> interaktivní režim</p> <pre># tar -cvf ssh.tar /etc/ssh (vytvoří archiv z daného adresáře) # tar -cvf scripts.tar backup.sh restore.sh (vytvoří archiv z daných souborů) # tar -cMf /dev/st0 / -V system_bckp (vytvoří zálohu systému na pásku) # tar -tvf ssh.tar (vypíše obsah archivu) # tar -xvf ssh.tar (rozbalí archiv) pro archivaci a zároveň (de)komprimaci nástrojem „compress“: # tar -cvZf ssh.tar.Z /etc/ssh # tar -xvZf ssh.tar.Z pro archivaci a zároveň (de)komprimaci nástrojem „gzip“: # tar -cvzf ssh.tar.gz /etc/ssh # tar -xvzf ssh.tar.gz pro archivaci a zároveň (de)komprimaci nástrojem „bzip2“: # tar -cvjf ssh.tar.bz2 /etc/ssh # tar -xvjf ssh.tar.bz2 pro archivaci a zároveň (de)komprimaci nástrojem „xz“: # tar -cvJf ssh.tar.xz /etc/ssh # tar --remove-files -cvJf maillog_old.tar.xz maillog-20161225 maillog-20170102 # tar -xvJf ssh.tar.xz</pre>

Archivace a komprimace	
[<zdroj>  ] <b>cpio</b> > <cil>	<p>archivuje či obnovuje data, <b>-o</b> vytvoří archiv, <b>-d</b> vytvoří adresářový strom, <b>-i</b> rozbalí archiv, <b>-m</b> zachová čas poslední změny souborů, <b>-t</b> vypíše obsah archivu, <b>-v</b> podrobný výpis</p> <pre>\$ ls   cpio -ov &gt; archiv.cpio</pre> <p>(vytvoří archiv z obsahu pracovního adresáře)</p> <pre># find / -mtime -1 -type f -print   cpio -ov &gt; /dev/rmt0</pre> <p>(vytvoří archiv na pásku ze všech souborů, které byly změněny za poslední den)</p> <pre>\$ cpio -o &gt; /dev/fd0 &lt;- ' &lt;soubor1&gt; &lt;- ' &lt;soubor2&gt; &lt;- '</pre> <p><b>Ctrl+d</b> (vytvoří archiv na disketu interaktivním způsobem)</p> <pre># cpio -ivmd /\* &lt; /dev/rmt0</pre> <p>(rozbalí archiv z pásky)</p> <pre>\$ rpm2cpio setup-2.12.2-6.el8.noarch.rpm   cpio -tv</pre> <p>(vypíše soubory v balíčku)</p> <pre>\$ rpm2cpio setup-2.12.2-6.el8.noarch.rpm   cpio -idv</pre> <p>(rozbalí soubory z balíčku)</p>
<b>compress</b> <soubor ...>	<p>komprimuje soubor (adresář musí být nejdříve archivován do jednoho souboru nástrojem „tar“; původní zdroj odstraní), <b>-c</b> vypíše výstup na STDOUT a původní zdroj zachová, <b>-d</b> dekomprimuje soubor (obdoba příkazu „uncompress“), <b>-v</b> podrobný výpis</p>
<b>uncompress</b>   <b>compress -d</b> <soubor ...>.Z	<p>dekomprimuje soubor (původní zdroj odstraní), <b>-c</b> vypíše výstup na STDOUT a původní zdroj zachová (obdoba příkazu „zcat“), <b>-v</b> podrobný výpis</p> <pre>\$ zcat ssh.tar.Z   tar -x</pre>
<b>gzip</b> <soubor ...>	<p>komprimuje soubor (adresář musí být nejdříve archivován do jednoho souboru nástrojem „tar“; původní zdroj odstraní), <b>-&lt;n&gt;</b> úroveň komprese (nabývá hodnot 1-9, čím vyšší hodnota, tím vyšší komprese, výchozí hodnota je 6), <b>-c</b> vypíše výstup na STDOUT a původní zdroj zachová, <b>-d</b> dekomprimuje soubor (obdoba příkazu „gunzip“), <b>-v</b> podrobný výpis</p> <pre># gzip -c access.log &gt; access.log_2017-02-15.gz &amp;&amp; &gt; access.log</pre> <p>(komprimuje soubor, který uloží pod novým názvem a obsah původního souboru smaže)</p>
<b>gunzip</b>   <b>gzip -d</b> <soubor ...>.gz	<p>dekomprimuje soubor (původní zdroj odstraní), <b>-c</b> vypíše výstup na STDOUT a původní zdroj zachová, <b>-l</b> vypíše jméno archivu, jeho velikost v B před a po kompresi a poměr komprese v procentech, <b>-t</b> ověří celistvost komprimovaného souboru, <b>-v</b> podrobný výpis</p> <pre>\$ gunzip -cd ssh.tar.gz   tar -x</pre>

Archivace a komprimace	
<b>bzip2</b> <soubor ...>	komprimuje soubor (vyšší komprese než „gzip“; adresář musí být nejdříve archivován do jednoho souboru nástrojem „tar“; původní zdroj odstraní), <b>-&lt;n&gt;</b> úroveň komprese (nabývá hodnot 1–9, čím vyšší hodnota, tím vyšší komprese, výchozí hodnota je 6), <b>-k</b> zachová původní zdroj, <b>-d</b> dekomprimuje soubor (obdoba příkazu „bunzip2“), <b>-v</b> podrobný výpis
<b>bunzip2</b>   <b>bzip2 -d</b> <soubor ...>.bz2	dekomprimuje soubor (původní zdroj odstraní), <b>-c</b> vypíše výstup na STDOUT a původní zdroj zachová, <b>-k</b> zachová původní zdroj <b>-t</b> ověří celistvost komprimovaného souboru, <b>-v</b> podrobný výpis \$ <b>bzip2 -cd ssh.tar.bz2   tar -x</b>
<b>xz</b> <soubor ...>	komprimuje soubor (vyšší komprese než „bzip2“; adresář musí být nejdříve archivován do jednoho souboru nástrojem „tar“; původní zdroj odstraní), <b>-&lt;n&gt;</b> úroveň komprese (nabývá hodnot 0–9, čím vyšší hodnota, tím vyšší komprese, výchozí hodnota je 6), <b>-k</b> zachová původní zdroj, <b>-d</b> dekomprimuje soubor (obdoba příkazu „unxz“), <b>-v</b> podrobný výpis
<b>unxz</b>   <b>xz -d</b> <soubor ...>.xz	dekomprimuje soubor (původní zdroj odstraní), <b>-c</b> vypíše výstup na STDOUT a původní zdroj zachová, <b>-k</b> zachová původní zdroj, <b>-l</b> vypíše jméno archivu, jeho velikost před a po kompresi a poměr komprese, <b>-t</b> ověří celistvost komprimovaného souboru, <b>-v</b> podrobný výpis \$ <b>xz -cd ssh.tar.xz   tar -x</b>
<b>zip</b> <cíl> <zdroj ...>	komprimuje soubor či adresář (původní zdroj zachová), <b>-r</b> rekurzivně, <b>-e</b> pod heslem \$ <b>zip -r ssh.zip /etc/ssh</b>
<b>unzip</b> <soubor>.zip	dekomprimuje soubor (původní zdroj zachová), <b>-d</b> <adresář> určí cílový adresář pro rozbalení archivu, <b>-l</b> vypíše obsah archivu, <b>-t</b> ověří celistvost komprimovaného souboru

From:

<https://prompt.cz/> - Prompt.cz

Permanent link:

<https://prompt.cz/soubory-a-adresare>

Last update: 2025/07/08 11:40

