

Porovnávání souborů

Příkazy

diff, comm, cmp, patch

Soubory pro příklady: <http://pitbul.wz.cz/unix>

cmp

- cmp porovnává 2 soubory. Jestliže jsou shodné, nevypisuje nic. Jinak vypisuje při standardním nastavení číslo znaku a řádku prvního rozdílu. Také oznamuje shodnost jednoho souboru se začátkem souboru druhého. Byty a řádky jsou číslovány od 1.

Použití: `cmp [-l] [-s] file1 file2 [skip1] [skip2]`

l - vypíše číslo bytu (decimálně) a obě hodnoty (oktalově) pro každý rozdíl

s - nic nevypisuje, pouze předává návratovou hodnotu

skip1, skip2 - při porovnávání přeskočí zadaný počet bytů

Příklad:

rozdíl ve výpisu pro

`cmp file1.txt file2.txt`

`cmp -l file1.txt file2.txt`

comm

- comm čte 2 soubory (měly by být seřazeny) a vypisuje tři sloupce textu. V prvním jsou řádky obsažené pouze v prvním souboru, v druhém sloupci řádky obsažené pouze v druhém souboru a ve třetím sloupci řádky společné pro oba soubory.

Použití: `comm [-123] file1 file2`

- 1 - potlačí sloupec řádků unikátních pro file1**
 - 2 - potlačí sloupec řádků unikátních pro file2**
 - 3 - potlačí sloupec řádků společných oběma souborům**
- file1, file2 - jména souborů, - značí standardní vstup**

Příklad:

Určete řádky společné souborům file1.txt, file2.txt a file3.txt

comm

- comm čte 2 soubory (měly by být seřazeny) a vypisuje tři sloupce textu. V prvním jsou řádky obsažené pouze v prvním souboru, v druhém sloupci řádky obsažené pouze v druhém souboru a ve třetím sloupci řádky společné pro oba soubory.

Použití: `comm [-123] file1 file2`

- 1 - potlačí sloupec řádků unikátních pro file1
 - 2 - potlačí sloupec řádků unikátních pro file2
 - 3 - potlačí sloupec řádků společných oběma souborům
- file1, file2 - jména souborů, - značí standardní vstup

Příklad:

Určete řádky společné souborům file1.txt, file2.txt a file3.txt

`comm -12 file1.txt file2.txt|comm -12 - file3.txt`

diff

- diff porovnává dva soubory a na standardní výstup zapisuje seznam změn potřebných k získání file2 z file1.

Normální výstup obsahuje řádky podobající se příkazům editoru ed pro konverzi souboru file1 na file2

n1 a n3,n4

n1,n2 d n3

n1,n2 c n3,n4

kde n1 a n2 jsou řádky v file1, n3, n4 v file2

po každém tomto řádku následují řádky postižené v prvním souboru (značené <) a v druhém souboru (značené >)

Příklad: Ukázka normálního výstupu

diff file1.txt file2.txt

diff

Použití: `diff [-bitw] [-c | -e | -f | -h | -n | -u] file1 file2`

b - ignoruje úvodní mezery/tabulátory

i - ignoruje velikost písmen

w - ignoruje všechny mezery/tabulátory

c – vypisuje odlišné řádky (+ 3 okolní - kontextové) tak, že řádky odstraněné z file1 jsou označeny '-', řádky přidané do file2 jsou označeny '+', změněné řádky '!'

e - vytváří skript pro editor ed zajišťující vytvoření file2 z file1

Příklad: Vytvořte některý seznam změn (zmeny.txt) zajišťující konverzi file1.txt na file2.txt

diff

Použití: `diff [-bitw] [-c | -e | -f | -h | -n | -u] file1 file2`

b - ignoruje úvodní mezery/tabulátory

i - ignoruje velikost písmen

w - ignoruje všechny mezery/tabulátory

c – vypisuje odlišné řádky (+ 3 okolní - kontextové) tak, že řádky odstraněné z file1 jsou označeny '-', řádky přidané do file2 jsou označeny '+', změněné řádky '!'

e - vytváří skript pro editor ed zajišťující vytvoření file2 z file1

Příklad: Vytvořte některý seznam změn (zmeny.txt) zajišťující konverzi file1.txt na file2.txt

`diff file1.txt file2.txt>zmeny.txt`

diff

Porovnávání adresářů

diff [-bitw] [-c | -e | -f | -h | -n | -u] [-l] [-r] [-s] [-S name]
directory1 directory2

- Vypisuje změny pro soubory stejného jména obsažené v obou adresářích a jména souborů, které jsou pouze v jednom z adresářů

r - rekurzivně prochází podadresáře

s - oznamuje i identické soubory

S name - porovnává až od souboru name

patch

- patch čte soubor obsahující libovolný ze seznamů změn (normální, kontextový, pro editor ed – automaticky určuje) a změny aplikuje je na file. Defaultně čte změny ze standardního vstupu

Použití

patch [-b|NR] [-c | -e | -n | -u] [-d dir] [-D define] [-i patchfile] [-o outfile]
[-p num] [-r rejectfile] [file]

b - před modifikací souboru uloží záložní kopii (přípona .orig)

R - reverzní aplikace změn – z novějšího souboru vytvoří původní (nelze použít se skriptem pro editor ed)

c - interpretuje soubor změn jako kontextový seznam (diff -c)

e - interpretuje soubor změn jako ed skript

n - interpretuje soubor změn jako normální výstup diff

i patchfile - čte seznam změn ze souboru patchfile

o outfile - namísto modifikace file uloží výsledek do souboru outfile

patch

Příklad:

Aplikujte soubor zmeny.txt z předchozího příkladu na file1.txt, výsledek uložte do souboru file4.txt

patch

Příklad:

Aplikujte soubor zmeny.txt z předchozího příkladu na file1.txt, výsledek uložte do souboru file4.txt

```
patch -i zmeny.txt -o file4.txt file1.txt
```

Nyní libovolným způsobem porovnejte vzniklý soubor file4.txt se souborem file2.txt

patch

Příklad:

Aplikujte soubor zmeny.txt z předchozího příkladu na file1.txt, výsledek uložte do souboru file4.txt

```
patch -i zmeny.txt -o file4.txt file1.txt
```

Nyní libovolným způsobem porovnejte vzniklý soubor file4.txt se souborem file2.txt

```
cmp file2.txt file4.txt
```

... měly by být stejné