Unix-like Operating Systems

Commands: awk and sed.

Jan Trdlička

Czech Technical University in Prague, Faculty of Information Technology
Department of Computer Systems

Contents

sed

- What is the meaning of the `-n` option at command `sed`?
  - It suppresses automatic printing of pattern space (input lines).

```
cat -n /etc/passwd | sed ' '
```

```
cat -n /etc/passwd | sed -n ' '
```

- What is the meaning of the following `sed` commands?
  - `p` ... print the current pattern space (input lines),

```
cat -n /etc/passwd | sed 'p'
```

```
cat -n /etc/passwd | sed -n 'p'
```

  - `d` ... delete pattern space (input lines),

```
cat -n /etc/passwd | sed 'd'
```

  - `s` ... replace a portion matched regular expression.

```
ls -l / | sed 's/root/ROOT/g'
```

sed – print/delete

- How to print the first 10 lines of the file `/etc/passwd` by command `sed`?

```
cat -n /etc/passwd | sed -n '1,10p'
```

```
cat -n /etc/passwd | sed '11,$d'
```

- How to print the last 10 lines of the file `/etc/passwd` by command `sed`?

```
cat -n /etc/passwd | \
sed -n "$(($(wc -l </etc/passwd)-9)),\$p"
```

- How to print line of the file `/etc/passwd`, that represents info about an user account `"root"` (see man 5 passwd)?

```
sed -En '/^root:/p' /etc/passwd
```

- How to print lines of the file `/etc/passwd`, that don't represent info about an user account `"root"`?

```
sed  -E '/^root:/d' /etc/passwd
```

sed – text replacement

- Use the output of command `ls -l /etc`. How to replace the first character at line, which define the file type (d, -, l), by string (dir:, file:, link:).
  - ```
    ls -l /etc | \
    sed -E 's/^d/dir:/' | \
    sed -E 's/^-/file:/' | \
    sed -E 's/^l/link:/'
    ```
  - ```
    ls -l /etc | \
    sed 's/^d/dir:/;s/^-/file:/;s/^l/link:/'
    ```
  - ```
    cat types.sed
    s/^d/dir:/
    s/^-/file:/
    s/^l/link:/

    ls -l /etc | sed -Ef types.sed
    ```
    ```

sed – text replacement

- Use the output of command `ls -l /`. How to replace all strings `"root"` by strings `"ADMIN"`?

```
ls -l / | sed 's/root/ADMIN/g'
```

- Use the output of command `ls -l /`. How to print only the second column by the command `sed`?

```
ls -l / | \
sed -E 's/^([^ ][^ ]*) [ ]*([^ ]*) .*$/\2/'
```

awk

- Explain the behaviour of command `awk`.
- Example

```
ls -l /etc | \
  awk '
    BEGIN { print "Regular files and directories" ;
            f = 0;
            d = 0;
    }
    /^-/  { f++; print f, $0; }
    /^d/  { d++; print d, $0 ;}
    END   { print "Number of files:", f ;
            print "Number of directories:", d;
    }'
```

awk – patterns

- How to print the first 10 lines of the file `/etc/passwd` by command `awk`?

```
cat -n /etc/passwd | awk 'NR<=10'
```

```
cat -n /etc/passwd | awk 'NR==1,NR==10'
```

```
cat -n /etc/passwd | \
awk '{ if (NR <= 10) { print $0;} }'
```

- How to print the last 10 lines of the file `/etc/passwd` by command `awk`?

```
cat -n /etc/passwd | \
awk "NR > $(( $(wc -l </etc/passwd) - 10 ))"
```

awk – patterns

- How to print lines containing string `"root"` from the file `/etc/group` by command `awk`?

```
awk '/root/' /etc/group
```

- How to print lines not containing string `"root"` from the file `/etc/group` by command `awk`?

```
awk '!/root/' /etc/group
```

- Use output of command `getent passwd`.
- How to print lines that represent account info about users whose names are Jan or Peter or Eliska?
  - Trivial solution

```
getent passwd | \
  awk '/Jan/ ; /Peter/ ; /Eliska/'
```

awk – patterns

- How to print lines that represent account info about users whose names are Jan or Peter or Eliska?
  - Correct solution

```
getent passwd | \
 awk -F: \
  '$5 ~ /^Jan / ; $5 ~ /^Peter ; / $5 ~ /^Eliska /'
```

```
cat names.awk
#! /usr/bin/awk
BEGIN { FS=":" }
$5 ~ /^Jan /
$5 ~ /^Peter /
$5 ~ /^Eliska /
```

```
getent passwd | awk -f names.awk
```

awk – actions

- How to print the name and the size of every regular file in `/etc/`?

```
ls -l /etc | \
awk '/^-/ { printf("%s: %d B\n", $9,$5); }'
```

- How to print the user name and the path to home directory for every user that has account on this system?
  Hint: Use command `getent passwd`.

```
getent passwd | \
awk 'BEGIN {FS=":"; OFS="..."}{print $1,$6}'
```

- How to print the user name of user that has an account on this system and has the longest name (the 5th column)?

```
getent passwd | \
awk -F: '{ l=length($5);
           if (l>m) { m=l; u=$5; } }
         END{ print u; }'
```

awk – functions

- What is the meaning of the following `awk` functions?
  - `length(str)`
    ... length of the string `str`,
  - `substr(str, from, len)`
    ... the `len`-character substring of `str` starting at `from`,
  - `sub(ere,repl[,in])`
    ... replace the first substring matching regular expression `ere` in the string `in`, by the string the `repl`,
  - `gsub(ere,repl[,in])`
    ... replace all substring matching regular expression `ere`,
  - `split(str,arr[,fs])`
    ... split the string `str` into the array `arr` according to the separators array `fs`.

awk – arrays

- Assume that the file `m.txt` has the following contents:

```
1       2       3
8       7       9
3       7       2
```

- How to change the order of columns, such that the column i ($0 \leq i < n$) will be swap with column $(n - i)$ (reverse column order).

```
3       2       1
9       7       8
2       7       3
```

```
awk '{ for (i=NF; i>0; i--) printf("%d\t", $i);
       printf("\n"); }' m.txt
```

awk – examples

- Use output of command `ps -ef`. How to print info about processes which are running under the user `root`.

```
ps -ef | awk '/^ *root /'
```

```
ps -ef | nawk '{ if ($1 == "root") {print $0}; }'
```

awk – examples

- Use output of command `ps -eo user,rss,comm`. How to print the number of processes running under the effective identity "root" and the number of memory (RSS) allocated by these processes.
- Example of output

```
$> ps -eo user,rss,comm | awk -f ps1a.awk
root: nproc=85 rss=602264 KB
```

- Solution A

```
$> cat ps1a.awk
/^ *root /{
  nproc=nproc+1
  rss=rss+$2;
}
END {
  printf("root: nproc=%d rss=%d KB\n", nproc, rss)
}
```

awk – examples

- Solution B

```
$> cat ps1b.awk
BEGIN {
  user="root";
  nproc=0;
  rss=0;
}
{
  if ( $1 == user )
  {
    nproc=nproc+1;
    rss=rss+$2;
  }
}
END {
  printf("%s: nproc=%d rss=%d KB\n", user, nproc, rss)
  printf("\n");
}
```

awk – examples

- Use the output from the command `ps -eo user,rss,comm`. How to print the following information about the processes of user whose name is given?

  `user_name: nproc=number_of_running_processes rss=size_of_RSS_memory_alocatted_by_these_processes, list=list_of_these_processes`

- Example of output

```
$> ps -eo user,rss,comm | awk -f ps2.awk
root: nproc=83 rss=582952 KB
      proclist: sched, /usr/sbin/init, ...
```

awk – examples

- Solution A

```
> cat ps2.awk
BEGIN {
  user="root";
  nproc=0;
  rss=0;
}
{
  if ( $1 == user )
  {
    proclist[nproc]=$3;
    nproc=nproc+1;
    rss=rss+$2;
  }
}
END {
  printf("%s: nproc=%d rss=%d KB\n", user, nproc, rss)
  printf("         proclist: ");
  for (i=0; i<nproc; i++) {
    printf("%s, ", proclist[i]);
  };
  printf("\n");
}
```

awk – examples

- Solution B

```
> cat ps3.awk
BEGIN {
  nproc=0;  rss=0;
}
{
  if ( $1 == user )
  {
    proclist[nproc]=$3;
    nproc=nproc+1;
    rss=rss+$2;
  }
}
END {
  printf("%s: nproc=%d rss=%d KB\n", user, nproc, rss)
  printf("         proclist: ");
  for (i=0; i<nproc; i++) {
    printf("%s, ", proclist[i]);
  };
  printf("\n");
}
```

```
$> ps -eo user,rss,comm | nawk -v user=root  -f ps3.awk
```