Linux Utilities

gawk

- *gawk* is a very powerful report generator and data management tool.
- It offers the following facilities:-
    - * Scanning of input data for patterns or relationships.
    - * Printing of output fields in any order (the most common use for *gawk*), including constant strings of text.
    - * Translation of data delimiters, etc.
    - * Action on fields, both textual and arithmetic.
    - * Control statements such as those found in traditional programming languages.
- *gawk* treats input data in a similar way to *sort*, that is by records and fields.
- Default input data separators are newline for records and space or tab for fields, but these can be altered if required.
- What does 'gawk' stand for?

  It's the **GNU** version of a program written by Mr **A**ho, Mr **W**einberger and Mr **K**erninghan!
- The UNIX versions are known as *awk* and *nawk*.

Linux Utilities

gawk Introductory Example

- *gawk* is the most complex individual command we are going to introduce you to on this course.
- "You can do almost anything with *gawk*"
    * Yes you can, but there may be other better ways, using something such as Perl, for example.
- What does *gawk* actually do?
    * Reads input line by line.
    * Checks each line to see if it meets specified conditions, and if it does, performs specified actions, for example:

$ ***gawk '/^[^#]/{print "Host "$2" \t Address "$1}' \ /etc/hosts***

```
Host didcot     Address 192.168.200.2
Host carlisle   Address 192.168.200.3
Host ash        Address 192.168.200.4
Host hunt       Address 192.168.200.5
Host gatwick    Address 192.168.200.6
Host cod        Address 192.168.200.7
Host perch      Address 192.168.200.8
Host stanstead  Address 192.168.200.9
Host golfer-gw  Address 192.168.200.10
etc....
```

- Takes all lines in the file */etc/hosts* which are not comments, and displays the workstation name and the Internet (IP) address.

Linux Utilities

gawk Introductory Example (contd.)

- Let's look at this in more detail.
- The SECOND parameter (easiest one first!) is the input file; in this example, we have chosen */etc/hosts* which is the file that defines the names and addresses of all workstations known on the network.
    * If no input file is given, then *gawk* reads from *stdin* - ideal for piping information in from another tool.
- The FIRST parameter contains the *gawk* statements.
- These statements, as in any programming language, tell *gawk* what actions to perform, and are known as the *gawk* program.
    * *gawk* programs can be quite complex ...

      /^[^#]/{print "Host "$2" \t Address "$1}
    * *gawk* programs are almost always going to need single quote shell protection (because they almost always contain $ characters) and will probably also contain many other special characters as well - even our first example contains space, [,], {, }, ", and $.

Linux Utilities

gawk Introductory Example (contd.)

- The *gawk* program consists of one or more statements of the form:

  pattern { action }

- In our example, the pattern was ***/^[^#]/*** - a regular expression!
- (The regular expression characters are delimited by the "/" at each end.)
- Specifically, it calls for the following action to be taken on all input lines which do not start with a # character - i.e. which are not comment lines in file */etc/hosts*.
  - OTHER PATTERNS LATER
- Our sample action was ***print "Host "$2"\t Address"$1***

  This statement instructs *gawk* to print out
  - The constant text *Host*
  - The second field from the line just read
  - A tab (\t)
  - The constant text *Address*
  - The first field from the line just read
  - Finally a new line character will be output
- As you may guess, this output is routed to '*stdout*'

Linux Utilities

Simple gawk example

- *gawk* has a number of sensible defaults, so you can write very effective commands much shorter than our first example
- If no action is given, lines that match are sent to stdout unchanged.
    - Example:

```
$ ls -l utilities | gawk '$5>2000'
-rw-r--r--  1 sa2    other     33864       Jan 17  1995 datafile_full
-rw-r--r--  1 sa2    other     21346       Jan 17  1995 datafile_part
-rw-r--r--  1 sa2    other     56394       Nov 22  1993 ex_data1
-rw-r--r--  1 sa2    other     56234       Nov 22  1993 ex_data2
-rw-r--r--  1 sa2    other     57034       Nov 22  1993 ex_data3
-rw-r--r--  1 sa2    other      2051       Nov 22  1993 ex_data4
etc.

$
```

- Uses *gawk* to print details of those files in the *utilities* directory which are larger than 2000 bytes in size.
- All lines output by the *ls* command are processed, as no pattern matching was applied.

Linux Utilities

gawk - printing selected fields

- As we saw in the initial example, it is easy to define the fields to be printed.
- Here is a further example, taking input from an *ls -l* command:-

$ ***ls -l /etc | gawk '{print "file "$9"\t size "$5}'***

```
file     size
file CORBA       size 4096
file DIR_COLORS  size 2434
file Muttrc      size 80316
file TextConfig  size 45212
file X11         size 4096
file a   size 0
file adjtime     size 12
file aliases     size 1023
file aliases.db  size 12288
file amanda      size 4096
file amandates   size 0
file amd.conf    size 688
file amd.net     size 105
file anacrontab  size 370
etc
$
```

* A TAB character (\t) was printed before the word "size", but was not 100% successful lining up the columns.
* The *printf* statement (later!) can be used instead of *print* to achieve better formatted output.

Linux Utilities

Exercise

- Use *gawk* to print the filename, owner and size from an *ls -l* listing.
- Use *gawk* to print out a table from the file *datafile_part*, (under the *utilities* directory) showing:
    * User name
    * Workstation name
    * Date logged in
    * Length of login

Linux Utilities

gawk - Arithmetic Operations

- *gawk* has a calculation capability
    - * Arithmetic calculations can be written straight into a *print* action ...

$ ***ls -l | gawk '{print $9,"\t size ",$5/1024,"Kb"}'***

```
Christian        size  0.0302734 Kb
Surnames         size  0.0351562 Kb
datafile_full    size  33.0703 Kb
datafile_part    size  20.8457 Kb
demo.final       size  0.368164 Kb
demo.new         size  0.328125 Kb
demo.orig        size  0.324219 Kb
ex_data1         size  55.0723 Kb
ex_data2         size  54.916 Kb
etc..
$
```

- Arithmetic operators include
    - \+ add
    - \- subtract
    - * multiply
    - / divide
    - % remainder when divided by
    - ( ) for changing order of precedence
- Other function such as *sqrt*, *sin*, *cos* and *log* are also available within *gawk*, as is a random number generator *rand*.
- A list of functions appears in the *gawk* manual page.