

Swiss Army Chainsaw

Powertool die in je broekzak past



Perl

- Programma bestanden
- Command line



Programmeren

- Iteratief
- Object georiënteerd
- Uitgebreide verzameling ontwikkeltools
 - Test framework
 - Coverage (Code, Documentatie)



Command line

- algemene scripts
- tools nadoen
 - grep
 - sed
 - awk
 - wc



Command line scripts

Command line scripts hebben altijd `-e` of `-E` (voor perl `>= 5.10.0` met features)

```
$ perl -E 'say ~~localtime'  
Sat Mar 5 10:30:38 2011
```



perl -n

```
perl -ne '<your-program>':
```

```
LINE:
```

```
    while (<>) {  
        # <your-program>  
    }
```



grep

- grep
 - `perl -ne '/regex/ && print' [filename]`
- `grep -H`
 - `perl -ne '/regex/ && print "$ARGV: $_" [filename]`
- `grep -n`
 - `perl -ne '/regex/ && print "$.: $_" [filename]`
- `grep -v`
 - `perl -ne '/regex/ || print" [filename]`



perl -p

```
perl -pe '<your-program>'
```

LINE:

```
while ( <> ) {  
    # <your-program>  
}  
continue {  
    print or die $!;  
}
```



sed

- `sed -e 's///'`
 - `perl -pe 's/pattern/replacement/' [filename]`
- `sed -e 'y///'`
 - `perl -pe 'y/searchlist/replacementlist/' [filename]`



perl -i -[np]e '...'

- -i
 - inplace editing (like sed)
- -i.bak
 - rename to <original file>.bak
 - create new <original file>
 - edit



perl -naF':'

- perl -na
 - @F = split;
- perl -naF':'
 - @F = split /:/, \$_;



awk

- `awk -F':' -e '{print $1}' /etc/passwd`
 - `perl -naF':' -le 'print $F[0]' /etc/passwd`



Soms is Perl eenvoudiger

- `sed -e '1!G;h;$!d' <file>`
 - `perl -e 'print reverse <>' <file>`



Soms ook niet...

```
perl -nae\  
  '$s{$ARGV}{lines}++;  
  $s{$ARGV}{words}+=@F;  
  $s{$ARGV}{bytes}+=length($_);  
  END{  
    printf(  
      "%8u%8u%8u %s\n",  
      @{$s{$_}}{qw/lines words bytes/},  
      $_  
    ) for keys %s;  
  }' \  
  /etc/passwd
```





Bedankt voor de aandacht

