

Unix et Programmation Shell

Philippe Langevin

département d'informatique
UFR sciences et technique
université du sud Toulon Var

Automne 2013

brouillon en révision

- site du cours :
<http://langevin.univ-tln.fr/cours/UPS/upsh.html>
- localisation du fichier :
<http://langevin.univ-tln.fr/cours/UPS/doc/man.pdf>

dernières modifications

upsh.tex	2024-01-28	21:05:32.653614398	+0100
tools.tex	2024-01-28	21:05:32.653614398	+0100
term.tex	2024-01-28	21:05:32.651614387	+0100
syntaxe.tex	2024-01-28	21:05:32.650614381	+0100
shell.tex	2024-01-28	21:05:32.649614376	+0100
prologue.tex	2024-01-28	21:05:32.648614371	+0100
proc.tex	2024-01-28	21:05:32.646614360	+0100
pipe.tex	2024-01-28	21:05:32.645614355	+0100
perm.tex	2024-01-28	21:05:32.645614355	+0100
man.tex	2024-01-28	21:05:32.643614344	+0100
part.tex	2024-01-28	21:05:32.644614349	+0100
file.tex	2024-01-28	21:05:32.642614338	+0100
direct.tex	2024-01-28	21:05:32.641614333	+0100
bash.tex	2024-01-28	21:05:32.639614322	+0100

2 - manuel

- man help
- navigation
- section
- piège
- a propos
- exercice
- format

documentation

On peut faire beaucoup de choses en mode texte :

commande [option-courte] [option-longue] [argument]

- filtrer : `grep`, `sed`, `find`
- compiler : `make`, `gcc`
- calculer : `bc`, `gmp`
- jouer : `gnuchess`
- naviguer : `lynx`, `links`
- etc. . .

Le manuel **unix** est un moyen efficace pour découvrir et retrouver les *options* des commandes populaires, les *prototypes* des bibliothèques, les *paramètres* de configuration.

pages du manuel

Il existe de nombreuses ressources pour obtenir de l'aide sur les commandes : livre, tutoriel, forum, moteur de recherche, howto, wiki. . .

`man` renseigne sur les commandes externes.

`help` renseigne sur les commandes internes.

Avant d'aller plus loin, rappelons comment sortir d'un mauvais pas sur la ligne de commande :

- CTRL-c : tuer la commande en cours
- CTRL-d : fermer l'entrée standard
- CTRL-z : stopper la commande
- CTRL-u : nettoyer la ligne de commande
- CTRL-r : retrouver une commande
- `readline`, `bind`, `.inputrc`

man page

Les composants **unix** sont documentés de manière concise sous forme de **man page**. Une information exploitable en ligne de commande par la commande **man**.

Raccourci	Action	Parm.
h	aide	
q	quitter	
flèches	navigation	
SPACE , b	changer de page	suivante, précédente
/?	rechercher	avant/arrière
n, N	occurrence	suivante, précédente

TABLE – commandes de **less**

La sortie du manuel est au format **roff** : un langage à balises léger et, par défaut, elle est interprétée par la commande **less**.

bien commencer : man man ...

```
~> man man | wc -l  
616
```

```
~> man man | col -b | grep -A2 AUTEUR  
AUTEUR
```

John W. Eaton est l'auteur historique de man. Federico Lucifredi <flucifredi@acm.org> en assure aujourd'hui la maintenance.

bien commencer : man man ...

```
↪ man man | col -b | grep BUGS -C4
```

A **manual page** consists of several sections.

Conventional **section** names include NAME, SYNOPSIS, CONFIGURATION, DESCRIPTION, OPTIONS, EXIT STATUS, RETURN VALUE, ERRORS, ENVIRONMENT, FILES, VERSIONS, CONFORMING TO, NOTES, BUGS, EXAMPLE, AUTHORS, and SEE ALSO.

section du manuel

- 1 commandes internes et externes.
- 2 appels système.
- 3 bibliothèque.
- 4 fichiers spéciaux.
- 5 formats des fichiers et conventions.
- 6 jeux.
- 7 divers (y compris les macropaquets et les conventions).
- 8 gestion du système.
- 9 Interface du noyau Linux.

man 1 intro

INTRO(1) Manuel de l'utilisateur Linux

NOM

intro – Introduction aux commandes
utilisateur

DESCRIPTION

La section 1 du manuel décrit les commandes et outils de l'utilisateur, comme les utilitaires de manipulation de fichiers, les interpreteurs de commandes, les compilateurs, les navigateurs web, les editeurs et outils de visualisation de fichiers et d'images, etc...

Exemple de boucle

```
↪ for x in {1..8}; do man $x intro | grep intro  
; done
```

```
intro – Introduction aux commandes utilisateur
```

```
intro – Introduction a la section des appels  
systeme
```

```
intro – Introduction aux fonctions de  
bibliotheque
```

```
intro – Introduction aux fichiers speciaux.
```

```
intro – Introduction a la section Formats de  
fichiers
```

```
intro – Introduction aux jeux
```

```
intro – Introduction a la section panoramas,  
conventions, et
```

```
intro – Introduction aux commandes d'
```

puzzle

```
1 char *p="char *p=%c%s%c;  
2 main(){printf (p,34, p,34);} "  
3 main(){printf(p,34, p,34);}
```

quine.c

puzzle

```
1 char *p="char *p=%c%s%c;  
2 main(){printf(p,34,p,34);}";  
3 main(){printf(p,34,p,34);}
```

quine.c

```
~> tr '\n' ' ' < quine.c > q.c  
~> cat q.c
```

puzzle

```
1 char *p="char *p=%c%s%c;  
2 main(){printf(p,34,p,34);}";  
3 main(){printf(p,34,p,34);}
```

quine.c

```
~> tr '\n' ' ' < quine.c > q.c  
~> cat q.c
```

```
char *p="char *p=%c%s%c; main(){printf(p,34,p  
,34);}"; main(){printf(p,34,p,34);}
```

puzzle

```
1 char *p="char *p=%c%s%c;  
2 main(){printf (p,34, p,34);}";  
3 main(){printf(p,34, p,34);}
```

quine.c

```
~> tr '\n' ' ' < quine.c > q.c  
~> cat q.c
```

```
char *p="char *p=%c%s%c; main(){printf (p,34, p  
,34);}"; main(){printf(p,34, p,34);}
```

```
~> gcc q.c  
./a.out  
char *p="char *p=%c%s%c; main(){printf (p,34, p  
,34);}"; main(){printf(p,34, p,34);}
```


hint

Trois consultations du manuel permettent de comprendre ce qui se passe :

- `man tr` pour obtenir des détails sur le filtre `tr`.
- `man printf` pour le mode d'emploi de la fonction `printf`.
- `man ascii` pour un mémo sur le codage des caractères.

man ascii

```
↪ man ascii | grep '30 40' -A12 | cut -c26-
```

```
30 40 50 60 70 80 90 100 110 120
```

0:	(2	<	F	P	Z	d	n	x	
1:)	3	=	G	Q	[e	o	y	
2:	*	4	>	H	R	\	f	p	z	
3:	!	+	?	I	S]	g	q	{	
4:	"	,	@	J	T	^	h	r		
5:	#	-	A	K	U	_	i	s	}	
6:	\$.	B	L	V	'	j	t	~	
7:	%	/	9	C	M	W	a	k	u	DEL
8:	&	0	:	D	N	X	b	l	v	
9:	'	1	;	E	O	Y	c	m	w	

printf du C??

~> man printf

PRINTF(1)

User Commands

NAME

printf — format and print data

SYNOPSIS

printf FORMAT [ARGUMENT]...

printf OPTION

DESCRIPTION

Print ARGUMENT(s) according to FORMAT, or execute according to OPTION:

—help display this help and exit

printf du C!

```
↪ man 3 printf
```

PRINTF(3)
Manual

Linux Programmer's

NAME

printf, fprintf, sprintf, snprintf, vprintf, vfprintf, vsprintf, vsnprintf – formatted output conversion

SYNOPSIS

```
#include <stdio.h>
```

```
int printf(const char *format, ...);
```

exercice

Utiliser le manuel pour expliquer :

```
↪ man man | col -b | grep -E '^[A-Z]+$'
```

NAME

SYNOPSIS

DESCRIPTION

EXAMPLES

OVERVIEW

DEFAULTS

OPTIONS

ENVIRONMENT

FILES

HISTORY

```
↪ man man | col -b | grep -Ex '[A-Z]+'
```

↪ man man | grep EXAMPLES -A24

EXAMPLES

man ls

Display the manual page for the item ls.

man -a intro

Display, in succession, all of the available

man -k printf

Search the short descriptions and manual page

names for the keyword printf as regular expression. Equivalent to **apropos** -r printf

man -f smail

Lookup the manual pages referenced by smail and print out the short descriptions of any found. Equivalent to **whatis** -r smail.

man -k RE

```
~> man -k '[a-z]{3} printf' | cut -c1-40
vasprintf (3)          - print to allocate
vfwprintf (3)         - formatted wide-ch
vfwprintf (3p)        - wide-character fo
vsprintf (3)          - formatted output
vsprintf (3p)         - format output of
vswprintf (3)         - formatted wide-ch
vswprintf (3p)        - wide-character fo
```

man -k RE

```
~> man -k '[a-z]{3} printf' | cut -c1-40
vasprintf (3)          - print to allocate
vfwprintf (3)         - formatted wide-ch
vfwprintf (3p)        - wide-character fo
vsnprintf (3)         - formatted output
vsnprintf (3p)        - format output of
vswprintf (3)         - formatted wide-ch
vswprintf (3p)        - wide-character fo
```

```
~> man -k '[a-z]{3} printf' | sed 's/-.*/@/'
vasprintf (3)          @
vfwprintf (3)         @
vfwprintf (3p)        @
vsnprintf (3)         @
vsnprintf (3p)        @
```


printf

```
↪ man -f printf
printf (3p)      - print formatted output
printf (3)       - formatted output conversion
printf (1)       - format and print data
printf (1p)      - write formatted output
```

- `whatis printf`

Il y a donc

- une fonction `printf`
- une commande `printf`

attention !

~> /bin/printf

/bin/printf: operande manquant

Saisissez /bin/printf —help pour plus
d'informations.

~> cd /bin

~> printf

printf: utilisation: printf [-v var]
format [args]

~> ./printf

./printf: operande manquant

Saisissez ./printf —help pour plus
d'informations.

piege classique

Les commandes homonymes font que le manuel ne renseigne pas toujours sur la commande que l'on croit !!!

echo

~> man echo

ECHO(1) User Commands

NAME

echo – display a line of text

SYNOPSIS

echo [SHORT-OPTION]... [STRING]...

echo LONG-OPTION

DESCRIPTION

Echo the STRING(s) to standard output.

–n do not output the trailing newline

–e enable interpretation of backslash

–E disable interpretation of backslash

—help display this help and exit

...

help

```
~> echo —help  
—help
```

```
~> which echo  
/bin/echo
```

```
~> /bin/echo —help
```

```
Utilisation : /bin/echo [SHORT-OPTION]...  
             ou : /bin/echo LONG-OPTION
```

```
...
```

```
~> type -f echo  
echo est une primitive du shell
```

help

↪ help echo

```
echo: echo [-neE] [arg ...]
      écrit les arguments sur la sortie standard.
```

```
Affiche les ARGs sur la sortie standard,
      suivis d'un retour a la ligne.
```

Options :

- n ne pas ajouter de saut de ligne
- e active l'interprétation des barres
contre-obliques d'échappement ci-dessous
- E supprime explicitement l'interprétation
les barres contre-obliques d'échappement

enable

```
~> echo —help
```

```
—help
```

```
~> enable -n echo
```

```
~> echo —help
```

```
Utilisation : echo [SHORT-OPTION [STRING]
```

```
ou : echo LONG-OPTION
```

enable, head, tail

```
~> enable | head -7  
enable .  
enable :  
enable [  
enable alias  
enable bg  
enable bind  
enable break
```


enable, head, tail

```
~> enable | head -7
enable .
enable :
enable [
enable alias
enable bg
enable bind
enable break
```

```
~> enable | tail -3
enable unalias
enable unset
enable wait
```

liste des commandes internes

```
↪ enable | sed 's/^[^ ]*//'| tr '\n' ' ' | fold -sw50  
. : [ alias bg bind break builtin caller cd  
command compgen complete compopt continue  
declare  
dirs disown echo enable eval exec exit export  
false fc fg getopts hash help history jobs kill  
let local logout mapfile popd printf pushd pwd  
read readarray readonly return set shift shopt  
source suspend test times trap true type typeset
```

liste des commandes internes

```
→ enable | sed 's/^[^ ]*//'|tr '\n' ' '|fold -sw50  
. : [ alias bg bind break builtin  
  caller  
  cd command compgen complete compopt  
continue declare dirs disown echo enable  
eval exec exit export false fc fg getopt  
hash help history jobs kill let local  
logout mapfile popd printf pushd pwd read  
readarray readonly return set shift shopt  
source suspend test times trap true type  
typeset ulimit umask unalias unset wait
```

exercice

Utiliser l'aide en ligne de commande pour expliquer la différence entre :

```
~> echo -E '\retou\rcha\rriot '
```

et

```
~> echo -e '\retou\rcha\rriot '
```

Utiliser le manuel pour comprendre les lignes :

```
↪ find ~/CC -name "*.c" -exec wc -l {} \; foo \  
  | awk 'BEGIN {s=0} {s=s+$0} END {print s}'
```

iliv ?

```
~> man iliv
```

iliv ?

```
~> man iliv
```

ILIV(6)

NAME

iliv – unix and shell programs

SYNOPSIS

```
iliv [-h|-a] [-lse] [-v type] [ list ]
```

DESCRIPTION

iliv provides informations on the introduction to GNU/linux system and Bash programs course

```
.\ " exemple de page
.\ " cours I54 USTV 2012
.TH ILIV 6 "falling 2012" "I54 Manual"
.SH NAME
iliv \- unix and shell programs
.SH SYNOPSIS
iliv
[\fB-h\fP] [\fB-a\fP] [\fB-lse\fP] [\fB-v\fP \fIttype\fP] [ list ]
.SH DESCRIPTION
iliv provides informations on the introduction to GNU/linux system and Bash programs courses.
.SH OPTIONS
The following options are supported
.TP
\fB-a\fP \fB-h\fP displays help
.TP
\fB-l\fP lecturer
.TP
\fB-s\fP slides
.TP
\fP-e\fP ecue
.SH LICENCE
Copyleft (!c) 2012 by Pavle Michko.
.SH AUTHOR
michko@euphoria.fr
.SH SEE ALSO
Joseph (kernel) Emmanuel (tcp/ip)
.SH BUG
please, report bugs administrative authority.
```


format ROFF

Les pages du manuels sont écrites dans le langage de formatage **roff**. Un langage à balises plus léger que \LaTeX utilisé pour les pages du manuel.

```
~> groff -man -Tascii iliv.6
```

si OK alors

```
~> cp iliv.6 /usr/share/man/.
```

Pour en savoir plus,

```
~> man 7 roff
```