isec Engenharia

Sistemas Operativos

Engenharia Informática - DEIS

2021/22

Comandos Unix e Shell **Ficha de apoio a exame (parte prática apenas)**

Lista não exaustiva de comandos Unix

cat	[-n] file1 ...	# Apresenta (opcionalmente concatenando) ficheiros
cd	[directory]	# Muda de directoria
chmod	[-R] {ugo+=-rwx} dir/file	# Muda permissões de ficheiros e directorias
chown	[-R] user.group dir/file	# Muda dono e grupo de ficheiros
cut	[-c list|-f list|-dsep] file	# Apresenta partes (colunas) de um ficheiro de texto
date	*formato*	# Apresenta a data e hora do sistema
diff	[-uNriw]	# Compara ficheiros e evidencia as diferenças
echo	[-n]	# Apresenta texto no *stdout*
find	[dir] [expression]	# Procura ficheiros recursivamente
grep	[-n] 'pattern' file	# Apresenta o conteúdo do ficheiro filtrando as linhas segundo uma expressão regular
groupadd	group	# Cria um grupo
groupdel	group	# Apaga um grupo
head	[-n] file	# Apresenta linhas iniciais de um ficheiro no *stdout*
id	[username]	# Apresenta a identificação do utilizador
ls	[-alR] [directory]	# Apresenta a lista de ficheiros e directorias
ln	[-s] target link	# Cria uma ligação para um ficheiro
man	command	# Apresenta as páginas de manual
mkdir	[-p|-m mode] directory	# Cria directorias
more	file	# Apresenta o conteúdo de um ficheiro com pausas
mv	file1 file2	# Move e renomeia ficheiros
nano	[+n] file	# Editor de texto simples baseado no editor pico (*Pine Composer*)
passwd	[-lu] [username]	# Modifica a password
printf	*Formatada args*	# Comando semelhante à função biblioteca C printf
pwd		# Apresenta o caminho para a directoria actual
read	var	# Obtém *input* via estrada standard (*stdin*)
rev	file1 ...	# copia o input para os ficheiros invertendo a ordem dos caracteres nas linhas
rm	[-ir] file	# Apaga ficheiros
rmdir	directory	# Apaga directorias
seq	[-fs] first last	# Produz sequências
sort	[-n|-tsep|-u] [+pos1] –k [n] fich	# Apresenta os dados *input* (conteúdo de ficheiro) por uma determinada ordem
su	[username]	# Adquire temporariamente outra identificação de utilizador (*substitute user*)
tac	file1 ...	# Apresenta ficheiro(s) invertidos (opcionalmente concatenando)
tail	[-n|+n] file	# Apresenta as linhas finais de um ficheiro no *stdout*
tee	[-a] fich1 ...	# Duplica (copia) a entrada (*stdin*) para *stdout* e ficheiros
tr	[-d] 'str1' ['str2']	# Substitui (*translate*) ou apaga caracteres do dados de entrada.
tree		# Mostra esquema de subdirectorias (comando tem que ser instalado)
umask	-S [ugo=+-rwx]	# Modifica a máscara default de permissões do utilizador
uname	[-r]	# Obtém o nome do sistema
uniq	[-idu]	# Identifica linhas com conteúdo repetido
useradd	[-c info] [-m] login	# Cria um utilizador
userdel	[-m]	# Apaga utilizador
wc	[-clw] file	# Conta caracteres, palavras e linhas
whereis		# Encontra ficheiros
who		# Indica que está *logado*

Exemplo de *output* de `ls -l`

```
-rw-rw-r-- 1 joao joao   357 Out  1  2015 so aulas praticas.txt
-rwxrwxr-x 1 joao joao  7380 Jul  8 16:56 teste1
```

Redireccionamento e Pipes

- comando **<** ficheiro — Redirecciona ficheiro para *stdin*
- comando **>** ficheiro — Redirecciona *stdout* para ficheiro
- comando **2>** ficheiro — Redirecciona *stderr* para ficheiro
- comando **>>** ficheiro — Redirecciona *stdout* para ficheiro (append)
- comando **>&** ficheiro — Redirecciona *stdout* and *stderr* para ficheiro
- comando1 **|** comando2 — Pipe: redirecciona o output(*stdout*) de comando 1 para input(*stdin*) de comando2

Caracteres com significado especial (nota: linha de comandos, que é diferente de sintaxe grep)

*	Zero ou mais caracteres
?	Um caracter

Gestão de utilizadores

/etc/passwd

```
# LOGIN:PASSWORD:UID:GID:REAL NAME:HOME DIR:COMMAND
jogos:x:7:13:jogos:/usr/jogos:
pcosta:x:1230:1050:Paulo Costa:/users/SO2007/pcosta:/bin/bash
manton:x:1167:1050:Maria Antonieta:/users/SO2007/manton:/bin/bash
```

/etc/group

```
# GROUP NAME:PASSWORD:GID:MEMBERS
bin:x:1:root,bin,daemon
so2015:x:1050:pcosta
rsi1516:x:1030:zezec,marcop
```

Shell Script

$0 ... **$9**	Parâmetros da linha de comandos
$*	Todos os parâmetros da linha de comandos
$#	Número de parâmetros da linha de comandos
$?	Código de retorno (*exit status*) do último comando executado
"..."	Protege uma *string*, mas reconhece $, \ e ` como especiais
'...'	Protege uma *string* (nenhum caracter é especial)
`...` **$(...)**	Executa comandos numa *subshell*
$((...))	Retorna o resultado de uma expressão aritmética
[...] **[[...]]**	Testa uma expressão

test expressão	Comparação numérica:	NUM1 **OP** NUM2	(**OP:** -lt, -gt, -le, -ge, -eq, -ne)
	Comparação de strings:	STR1 **OP** STR2	(**OP:** =, !=, -n, -z)
	Testes em ficheiros:	**OP** FICH	(**OP:** -e, -f, -d, -r, -w, -x)
expr expressão	Operações aritméticas	NUM1 **OP** NUM2	(**OP:** +, -, /, *)
$((expressão) **)**			

Estruturas de controlo *bash*:

```
if condição
then
  ...
elif condição
then
  ...
else
  ...
fi
```

```
while condição
do
  ...
done
```

```
until condição
do
  ...
done
```

```
for variável in lista
do
  ...
done
```

```
for ((exp1; exp2; exp3))
do
  ...
done
```

```
case variável in
  txt1) ...
        ;;
  txt2) ...
        ;;
  txtN) ...
        ;;
     *) ...
        ;;
esac
```