

SCIENCES SUP

Cours et exercices corrigés

1^{er} cycle/Licence • Écoles d'ingénieurs • IUT • BTS

UNIX, LINUX ET LES SYSTÈMES D'EXPLOITATION

2^e édition

Michel Divay

DUNOD

2-005-668-1



2-005-668-1

UNIX, LINUX ET LES SYSTÈMES D'EXPLOITATION

Cours et exercices corrigés

Michel Divay

Professeur à l'université Rennes 1

2^e édition

DUNOD

Table des matières

AVANT-PROPOS	XI
CHAPITRE 1 • UNIX	
1.1 Présentation générale d'un SE	1
1.2 Principe de la construction d'un SE	2
1.3 Le cas de MS-DOS	3
1.4 Historique, présentation générale d'Unix	5
1.5 Les commandes générales d'Unix	8
1.5.1 Généralités sur les commandes Unix	8
1.5.2 Systèmes de fichiers arborescents	8
1.5.3 Caractères génériques (métacaractères)	10
1.5.4 Compléments de noms	11
1.5.5 Droits d'accès	11
1.5.6 Commandes de base UNIX	13
1.5.7 Redirections, tubes	35
1.6 Le Shell	38
1.6.1 Les variables	39
1.6.2 Résultats d'une commande	41
1.6.3 Les expressions arithmétiques	42
1.6.4 Les scripts shell	43
1.6.5 Tableau, alias	58
1.7 Résumé	63
CHAPITRE 2 • LES FILTRES UNIX	
2.1 Paramètres d'une commande	65
2.1.1 Options de la ligne de commande	65
2.1.2 Variables d'environnement en langage C	67
2.1.3 Exemple : la nouvelle commande eko	68

2.2	Les filtres en C	71
2.2.1	Définition	71
2.2.2	Exemples de filtres écrits en C	72
2.3	Quelques filtres standard Unix	80
2.3.1	grep et les expressions régulières	80
2.3.2	Filtres tr, cut, colrm, sort, paste, uniq	82
2.3.3	Filtres nl, pr	85
2.3.4	Filtre sed (<i>stream editor</i>)	86
2.3.5	Filtre awk	86
2.4	Résumé	86
CHAPITRE 3 • LE SGF UNIX		
3.1	Les différents types de fichiers	87
3.2	Les répertoires du système	89
3.3	Les différents types de systèmes de fichiers	91
3.4	Structure générale d'un sgf	91
3.4.1	Inode	91
3.4.2	Implantation des données d'un fichier	92
3.5	Répertoire	95
3.6	Les tables d'occupation	97
3.7	Le découpage en groupes	98
3.8	Répertoires et sous-répertoires	100
3.9	Système de fichiers de type ext2	102
3.9.1	Système de fichiers (vide) après mkfs	103
3.9.2	Création de fichiers et répertoires	106
3.10	Variantes pour la gestion des inodes et des blocs libres	107
3.10.1	Allocation, désallocation d'inodes (variante)	108
3.10.2	Allocation, désallocation de blocs de données (variante)	109
3.11	Définition des appels système SGF	110
3.12	Primitives d'entrées-sorties sous Unix	111
3.12.1	Description des appels système concernant les entrées-sorties	114
3.12.2	Exemple de programme avec appels système du SGF	126
3.12.3	fentl : contrôle des caractéristiques d'un fichier ouvert	133
3.12.4	Verrouillage de fichiers	134
3.13	Bibliothèque d'entrées-sorties	146
3.13.1	Principe	146
3.13.2	La bibliothèque stdio.h	146
3.13.3	Exemple de création d'une bibliothèque d'entrées-sorties	150
3.14	Le buffer cache (entrées-sorties en mode bloc)	157
3.15	Buffer cache (Linux)	164
3.16	Date et heure	167
3.17	Résumé	168

CHAPITRE 4 • LES APPELS SYSTÈME POUR LES PROCESSUS

4.1	Définitions	169
4.2	Les processus	170
4.3	Les appels système (processus)	171
4.3.1	Caractéristiques d'un processus	171
4.3.2	L'appel système pipe (tube anonyme)	173
4.3.3	L'appel système execve et les fonctions de la famille exec	174
4.3.4	L'appel système fork de création de processus	177
4.3.5	Autres appels système sur les processus	178
4.4	Utilisation de fork, exec, pipe, dup2	179
4.4.1	Exécution de deux processus séquentiels indépendants	179
4.4.2	Redirection des entrées et des sorties sur un fichier	180
4.4.3	Redirection des entrées et des sorties sur un tube	183
4.5	Interpréteur de commandes	189
4.6	Résumé	197

CHAPITRE 5 • PROCESSUS ET SÉMAPHORES

5.1	Introduction	199
5.2	Terminologie	199
5.3	Exclusion mutuelle	200
5.3.1	Problème : pourquoi ça ne marche pas ?	200
5.3.2	Exemple de principe de l'exclusion mutuelle	201
5.4	Exclusion mutuelle et sémaphores	202
5.4.1	Définitions	202
5.4.2	Réalisation d'opérations P et V avec des tubes (pipe Unix)	202
5.4.3	Principe de l'exclusion mutuelle avec sémaphore	204
5.4.4	Exemple du carrefour routier	204
5.4.5	Changement d'états d'un processus	207
5.4.6	Exemple des baigneurs	208
5.5	Synchronisation avec des sémaphores	216
5.5.1	Définition	216
5.5.2	Exemple des moyens de transport	217
5.5.3	Exemple du producteur-consommateur	220
5.6	Interblocage (deadlock)	223
5.7	Résumé	224

CHAPITRE 6 • COMMUNICATION ENTRE PROCESSUS

6.1	Communiquer : du simple au complexe	225
6.2	Les signaux	226
6.3	Principe du client-serveur	237
6.4	Les tubes nommés ou FIFO	238
6.4.1	Exemple de communication par tube nommé	239
6.4.2	Exemple des baigneurs	241
6.4.3	Client-serveur avec des tubes nommés	241

6.5	Les files de messages	242
6.5.1	Définitions	242
6.5.2	Exemple : menu de test des files de messages	244
6.5.3	Module sur les files de messages	249
6.6	Client serveur avec files de messages	249
6.6.1	Le principe	249
6.6.2	Les clients : étudiant, enseignant	251
6.6.3	Communication par file de messages	256
6.7	Les sémaphores Unix	260
6.7.1	Définitions	260
6.7.2	P et V avec les sémaphores Unix	261
6.8	Les segments de mémoire partagée	266
6.8.1	Définitions	266
6.8.2	Producteur-consommateur (tampon 1 place)	267
6.8.3	Producteur-consommateur (tampon N places)	268
6.9	Synchronisation et mémoire partagée : les 5 philosophes	270
6.9.1	Première solution : fausse ; il peut y avoir interblocage	271
6.9.2	La solution correcte	271
6.10	Application avec mémoire partagée et sémaphores : bataille navale	277
6.11	Résumé	283
 CHAPITRE 7 • LES THREADS UNIX		
7.1	Introduction	285
7.2	Les fonctions concernant les threads	285
7.3	Les fonctions concernant les MUTEX	287
7.4	Exemple de synchronisation de threads	288
7.5	Exemple de l'éditeur de texte	292
7.6	Les conditions et les moniteurs	296
7.6.1	Principe du moniteur	296
7.6.2	Exemple de la synchronisation des moyens de transport	297
7.6.3	Réalisation avec les Pthreads	298
7.7	Diverses remarques sur les threads	304
7.8	Résumé sur les threads	306
 CHAPITRE 8 • LES SOCKETS		
8.1	Définition des sockets	307
8.2	Fonctions utilitaires (adresses IP, ports, tampons)	308
8.2.1	Les conversions d'entiers de format Internet en format ordinateur hôte	308
8.2.2	Les fonctions utilitaires de copie ou de comparaison de zones mémoires	309
8.2.3	Les structures et fonctions concernant les adresses IP	309
8.2.4	Les services prédéfinis	314
8.2.5	Programme de test des fonctions sur les adresses et les ports	314

8.3	Les fonctions de communication en mode UDP	317
8.3.1	Les appels système pour la communication UDP	317
8.3.2	Client-serveur	321
8.3.3	Le module des opérations sur les sockets	323
8.3.4	Client-serveur d'enchères en mode UDP	326
8.4	Les fonctions de communication en mode TCP	333
8.4.1	Les appels système pour la communication TCP	333
8.4.2	Client-serveur avec des sockets TCP	339
8.5	Serveur avec ou sans états (statefull ou stateless)	345
8.6	Résumé	346
CHAPITRE 9 • GESTION DE LA MÉMOIRE ET ORDONNANCEMENT		
9.1	Relocation des programmes	347
9.2	Gestion de la mémoire : les variantes	348
9.2.1	Un seul utilisateur en mémoire en plus du SE	348
9.2.2	Plusieurs utilisateurs plus le SE en mémoire centrale	348
9.2.3	Segmentation	350
9.2.4	Pagination et mémoire virtuelle	351
9.3	Exemple : Linux	354
9.3.1	Adresses virtuelles	354
9.3.2	Adresses réelles	357
9.4	Ordonnancement (scheduling)	372
9.4.1	Mémorisation des processus	372
9.4.2	État d'un processus, files d'attente	375
9.4.3	Scheduler et priorités	375
9.4.4	Arbre des processus	377
9.5	Résumé	378
9.6	Conclusion générale	378
CORRIGÉS DES EXERCICES		379
BIBLIOGRAPHIE		423
INDEX		425