

Table des matières

Préface	iii	2.2 Les pilotes de périphériques	19
PREMIÈRE PARTIE :		2.3 Logiciel d'entrée-sortie indépendant du matériel	19
PRINCIPES DE CONCEPTION DES SYSTÈMES D'EXPLOITATION	1	2.4 Logiciels d'entrée-sortie faisant partie de l'espace de l'utilisateur	21
Chapitre 1 Structure d'un système d'exploitation ..	3	Chapitre 3 Le système Linux étudié	23
1 Les trois grandes fonctions	3	1 Le système Linux à étudier	23
1.1 Chargement des programmes	3	1.1 Noyau et distribution	23
1.2 Le système d'exploitation en tant que machine virtuelle	4	1.2 Noyau minimal	23
1.3 Le système d'exploitation en tant que gestionnaire de ressources	4	1.3 Obtention des sources	24
2 Caractéristiques d'un système d'exploitation	5	1.4 Programmation Linux	24
2.1 Systèmes multi-tâches	5	1.5 Versions du noyau Linux	24
2.2 Systèmes multi-utilisateurs	7	2 Les sources du noyau 0.01	25
3 Structure externe d'un système d'exploitation	9	2.1 Vue d'ensemble sur l'arborescence	25
3.1 Noyau et utilitaires	9	2.2 L'arborescence détaillée	25
3.2 Le gestionnaire de tâches	9	3 Vue d'ensemble sur l'implémentation	32
3.3 Le gestionnaire de mémoire	9	3.1 Caractéristiques	32
3.4 Le gestionnaire de fichiers	9	3.2 Étapes de l'implémentation	32
3.5 Le gestionnaire de périphériques	10	4 Évolution du noyau	34
3.6 Le chargeur du système d'exploitation	10	4.1 Cas du noyau 2.4.18	34
3.7 L'interpréteur de commandes	10	4.2 Aide au parcours du code source	35
4 Structure interne d'un système d'exploitation	10	4.3 Cas du noyau 2.6.0	35
4.1 Les systèmes monolithiques	11	DEUXIÈME PARTIE :	
4.2 Systèmes à modes noyau et utilisateur	11	UTILISATION DU MICRO-PROCESSEUR INTEL	37
4.3 Systèmes à couches	11	Chapitre 4 Prise en compte de la mémoire Intel ...	39
4.4 Systèmes à micro-noyau	12	1 La segmentation sous Intel	39
4.5 Systèmes à modules	13	1.1 Notion	39
5 Mise en œuvre	14	1.2 La segmentation en mode protégé sur Intel ..	39
5.1 Les appels système	14	2 La segmentation sous Linux	45
5.2 Les signaux	14	2.1 Mode noyau et mode utilisateur	45
Chapitre 2 Principe de traitement des entrées-sorties	15	2.2 Segmentation en mode noyau	46
1 Principe du matériel d'entrée-sortie	15	2.3 Accès à la mémoire vive	50
1.1 Les périphériques d'entrée-sortie	15	3 Évolution du noyau	51
1.2 Les contrôleurs de périphériques	16	3.1 Prise en compte d'autres micro-processeurs ..	51
1.3 Transferts synchrones et asynchrones	17	3.2 Accès à la mémoire vive	52
1.4 Périphériques partagés et dédiés	18	3.3 Utilisation de la segmentation	52
2 Principe des logiciels d'entrée-sortie	18	Chapitre 5 Adaptation des entrées-sorties et des interruptions Intel	55
2.1 Objectifs des logiciels des entrées-sorties ...	18	1 Accès aux ports d'entrée-sortie	55

1.1	Accès aux ports d'entrée-sortie sous 80x86 ..	55	1.13	Informations sur les fichiers utilisés	89
1.2	Encapsulation des accès aux ports d'entrée-sortie sous Linux	55	1.14	Table locale de descripteurs	90
2	Les interruptions sous Linux	56	1.15	Segment d'état de tâche	90
2.1	Rappels sur les vecteurs d'interruption d'Intel	56	2	Tâche initiale.....	94
2.2	Adaptations sous Linux	60	3	Table des processus	97
3	Initialisation des exceptions	61	3.1	Stockage des descripteurs de processus	98
3.1	Initialisation provisoire	61	3.2	Implémentation de la table des processus	99
3.2	Initialisation définitive	62	3.3	Repérage d'un descripteur de processus	99
4	Initialisation des interruptions matérielles.....	64	3.4	La tâche en cours	99
4.1	Un problème de conception	64	4	Évolution du noyau	100
4.2	Contrôleur d'interruptions programmable	65	4.1	Structure du descripteur de processus	100
4.3	Programmation des registres de contrôle d'initialisation du PIC	66	4.2	Table des processus	102
4.4	Programmation des registres de contrôle des opérations du PIC	68	Chapitre 7	Description du système de fichiers	103
4.5	Reprogrammation du PIC dans le cas de Linux	70	1	Étude générale	103
4.6	Gestionnaires des interruptions matérielles ...	71	1.1	Notion de fichiers	103
4.7	Manipulation des interruptions matérielles ...	72	1.2	Gestion des fichiers	104
5	Initialisation de l'interruption logicielle	72	1.3	Les fichiers du point de vue utilisateur	104
6	Évolution du noyau	73	1.4	La conception des systèmes de fichiers	106
6.1	Accès aux ports d'entrée-sortie	73	2	Caractéristiques d'un fichier.....	109
6.2	Insertion des portes d'interruption	73	2.1	Types de fichiers	109
6.3	Initialisation des exceptions	74	2.2	Droits d'accès d'un fichier sous UNIX	110
6.4	Initialisation des interruptions matérielles	76	2.3	Mode d'un fichier sous UNIX	111
6.5	Manipulation des interruptions matérielles ...	76	3	Notion de tampon de disque dur	111
TROISIÈME PARTIE :			4	Structure d'un disque MINIX.....	112
LES GRANDES STRUCTURES			4.1	Bloc sous MINIX et Linux	112
DE DONNÉES		79	4.2	Structure générale d'un disque MINIX	112
Chapitre 6 Les structures de données concernant les processus		81	4.3	Les nœuds d'information sur disque	113
1	Descripteur de processus.....	81	4.4	Le super bloc	115
1.1	Structure du descripteur de processus	81	5	Système de fichiers MINIX chargé en mémoire	116
1.2	Aspects structurels	82	5.1	Antémémoire	116
1.3	État d'un processus	82	5.2	Les descripteurs de nœud d'information	119
1.4	Priorité d'un processus	83	5.3	Table des super-blocs	121
1.5	Signaux	84	5.4	Les descripteurs de fichiers	122
1.6	Code de statut	85	6	Fichiers de périphériques.....	123
1.7	Espace d'adressage	85	6.1	Caractéristiques	123
1.8	Identificateurs du processus	85	6.2	Repérage des fichiers de périphériques	124
1.9	Hierarchie des processus	86	7	Évolution du noyau	124
1.10	Propriétaire d'un processus	87	7.1	Prise en charge de plusieurs systèmes de fichiers	124
1.11	Informations temporelles	88	7.2	Cas de POSIX	125
1.12	Utilisation du coprocesseur mathématique ...	89	7.3	Système de fichiers virtuel	125
			7.4	Super-bloc	126
			7.5	Nœud d'information	128
			7.6	Descripteur de fichier	129
			7.7	Répertoire	130
			7.8	Types de fichiers	131

7.9 Déclaration d'un système de fichiers	131	2.1 L'horloge temps réel des PC	194
7.10 Descripteur de tampon	131	2.2 Minuteur périodique programmable	196
Chapitre 8 Les terminaux sous Linux	133	3 Programmation du minuteur sous Linux.....	198
1 Les terminaux	133	3.1 Initialisation du minuteur	199
1.1 Notion de terminal	133	3.2 Variable de sauvegarde du temps	199
1.2 Les terminaux du point de vue matériel	133	3.3 Gestionnaire de l'interruption d'horloge	199
1.3 Le pilote de terminal	139	3.4 La comptabilisation du processus en cours ...	200
1.4 Les différents terminaux et les normes	139	4 Maintien de la date et de l'heure sous Linux.....	201
1.5 Modélisation en voies de communication	140	4.1 Variable structurée de conservation du temps	201
2 Paramétrage des voies de communication	140	4.2 Initialisation de la variable structurée	202
2.1 Principe	140	5 Évolution du noyau	204
2.2 La structure de paramétrisation	141	Chapitre 11 Le gestionnaire des tâches	209
2.3 Paramétrage des modes d'entrée	141	1 Commutation de processus	209
2.4 Paramétrage des modes de sortie	143	1.1 Notion générale	209
2.5 Le tableau des caractères de contrôle	145	1.2 Gestion du coprocesseur arithmétique	209
2.6 Paramétrage des modes locaux	148	1.3 Cas de Linux	209
2.7 Paramétrages des modes de contrôle	150	2 Ordonnancement des processus	210
3 Implémentation des voies de communication.....	151	2.1 Politique d'ordonnancement	211
3.1 Implémentation d'un tampon d'entrée ou		2.2 Algorithme d'ordonnancement	212
de sortie	151	3 Initialisation du gestionnaire des tâches	215
3.2 Implémentation des voies de communication .	153	4 Évolution du noyau	215
4 Implémentation du terminal.....	154	Chapitre 12 Les signaux sous Linux	221
4.1 Définition du terminal	154	1 Notion générale de signal	221
4.2 Les caractères de contrôle	155	2 Liste et signification des signaux	221
4.3 Caractéristiques de la console	155	3 Vue d'ensemble de manipulation des signaux	223
4.4 Caractéristiques des liaisons série	156	4 Implémentation des deux appels système	224
4.5 Les tampons du terminal	156	4.1 Implémentation de l'appel système d'envoi	
5 Évolution du noyau	156	d'un signal	224
		4.2 Implémentation de l'appel système de	
		déroutement	225
QUATRIÈME PARTIE :		5 Implémentation du traitement des signaux	225
ASPECT DYNAMIQUE SANS AFFICHAGE	163	6 Fonction de gestion de signal par défaut	227
Chapitre 9 Implémentation des appels système		7 Évolution du noyau	227
sous Linux	165	CINQUIÈME PARTIE :	
1 Principe.....	165	AFFICHAGE	231
1.1 Définition des appels système	165	Chapitre 13 Le pilote d'écran sous Linux	233
1.2 Notion de code d'erreur	168	1 Affichage brut	233
1.3 Insertion et exécution des appels système	169	1.1 Rappels sur l'affichage texte sur l'IBM-PC ...	233
1.4 Fonction d'appel	171	1.2 Implémentation sous Linux	234
2 Liste des codes d'erreur	172	2 Notion d'affichage structuré	237
3 Liste des appels système.....	174	2.1 Principe du logiciel d'affichage structuré	237
4 Évolution du noyau	178	2.2 Cas de Linux	238
Chapitre 10 Mesure du temps sous Linux	189	3 Les suites d'échappement ECMA-48	238
1 Les horloges	189	3.1 Syntaxe	238
1.1 Le matériel de l'horloge	190	3.2 Sémantique	239
1.2 Le logiciel des horloges	191		
2 Horloges matérielles des PC.....	194		

4	Le pilote d'écran sous Linux	240	2.5	Les macros auxiliaires	296
4.1	Prise en compte des caractéristiques ECMA-48	240	3	La routine int3()	296
4.2	Fonction d'écriture sur la console	241	4	La routine device_not_available()	297
4.3	Traitement des cas spéciaux	245	4.1	La routine principale	297
5	Évolution du noyau	254	4.2	La fonction math_state_restore() ..	298
5.1	Affichage graphique et affichage console	254	5	Évolution du noyau	298
5.2	Caractéristiques de l'écran	254	Chapitre 17 Mémoire virtuelle sous Linux		301
5.3	Les consoles	259	1	Étude générale	301
Chapitre 14 L'affichage des caractères sous Linux ...		263	1.1	Mémoire virtuelle	301
1	Traitement des caractères	263	1.2	Mise en place de la mémoire virtuelle	301
1.1	Les caractères	263	2	Pagination	302
1.2	Classification primaire des caractères	263	2.1	Notion	302
1.3	Fonctions de classification des caractères	264	2.2	Pagination à plusieurs niveaux	302
1.4	Fonctions de conversion	265	2.3	Protection	304
2	Écriture sur une voie de communication	266	3	La pagination sous Intel 80386	304
2.1	Description	266	3.1	Taille des pages	304
2.2	Implémentation	266	3.2	Structure des entrées des tables	305
2.3	Attente du vidage du tampon d'écriture	267	3.3	Activation de la pagination	306
2.4	Traitement des processus en attente	268	3.4	Structure d'une adresse virtuelle	306
3	Évolution du noyau	269	3.5	Mécanisme de protection matérielle	306
3.1	Traitement des caractères	269	4	La pagination sous Linux	306
3.2	Écriture sur une voie de communication	269	4.1	Mise en place des éléments	306
Chapitre 15 L'affichage formaté du noyau		273	4.2	Initialisation de la pagination	307
1	Nombre variable d'arguments	273	4.3	Zone fixe et zone de mémoire dynamique	308
1.1	L'apport du C standard	273	4.4	Structures de gestion des tables de pages	309
1.2	Implémentation de <i>stdarg.h</i> sous Linux ..	274	4.5	Obtention d'un cadre de page libre	310
2	Formatage	275	4.6	Libération d'un cadre de page	311
2.1	La fonction sprintf()	275	5	Traitement de l'exception de défaut de page	311
2.2	Structure des formats	275	5.1	Le code principal	312
2.3	Le cas de Linux 0.01	277	5.2	Exception d'essai d'écriture sur une page en lecture seule	313
2.4	Implémentation de vsprintf() sous Linux	277	5.3	Exception de page non présente	314
2.5	Les fonctions auxiliaires	280	6	Évolution du noyau	315
3	La fonction printf()	283	SEPTIÈME PARTIE :		
4	La fonction panic()	284	FICHIERS RÉGULIERS		325
5	Évolution du noyau	284	Chapitre 18 Le pilote du disque dur		327
SIXIÈME PARTIE :			1	Géométrie des disques durs	327
ASPECT DYNAMIQUE AVEC AFFICHAGE		289	1.1	Description générale	327
Chapitre 16 Gestionnaires des exceptions		291	1.2	Prise en charge par Linux	328
1	Traitement des exceptions sous Linux	291	2	Le contrôleur de disque dur IDE	330
2	Structure générale des routines	292	2.1	Les registres IDE	330
2.1	Définitions des gestionnaires	292	2.2	Les commandes du contrôleur IDE	334
2.2	Structure d'un gestionnaire	292	3	Prise en charge du contrôleur par Linux	341
2.3	Les fonctions de traitement du code d'erreur	293	3.1	Constantes liées au contrôleur	341
2.4	Les fonctions C des gestionnaires par défaut .	294			

3.2	Routine d'interruption matérielle du disque dur	342	4.3	Création d'un descripteur de tampon	377
3.3	Passage des commandes	344	4.4	Lecture d'un tampon	379
3.4	Fonction d'attente du contrôleur	345	5	Évolution du noyau	379
3.5	Récupération des erreurs	345	Chapitre 20 Les périphériques bloc	385	
4	Partitionnement du disque dur	346	1	Vue d'ensemble	385
4.1	Un choix d'IBM	346	2	Accès à bas niveau	386
4.2	Utilisation par Linux	348	2.1	Détermination des périphériques bloc	386
5	Requêtes à un disque dur	348	2.2	Table des pilotes de bas niveau	387
5.1	Notion de requête	348	2.3	Fonction d'accès à bas niveau	387
5.2	Structure des requêtes	348	3	Les fonctions de lecture et d'écriture de bloc	388
5.3	Tableau des listes de requêtes	349	3.1	Fonction d'écriture	388
5.4	Initialisation du disque dur	349	3.2	Fonction de lecture	389
5.5	Requête de lecture ou d'écriture	350	4	Évolution du noyau	391
5.6	Gestion des tampons	351	Chapitre 21 Gestion des nœuds d'information	395	
5.7	Ajout d'une requête	353	1	Chargement d'un super-bloc	395
5.8	Traitement des requêtes	355	2	Gestion des tables de bits des données	395
5.9	Le gestionnaire d'interruption en cas d'écriture	356	2.1	Recherche d'un bloc de données libre	395
5.10	Réinitialisation du disque dur	357	2.2	Macros auxiliaires	397
5.11	Le gestionnaire d'interruption en cas de lecture	359	2.3	Libération d'un bloc de données	398
6	Pilote du disque dur	359	3	Les fonctions internes des nœuds d'information	399
7	Évolution du noyau	360	3.1	Verrouillage d'un descripteur de nœud	399
7.1	Périphériques bloc	360	3.2	Déverrouillage d'un descripteur de nœud	399
7.2	Géométrie d'un disque dur	363	3.3	Fonction d'attente de déverrouillage	400
7.3	Initialisation d'un disque dur traditionnel	363	3.4	Écriture d'un nœud d'information sur disque .	400
7.4	Contrôleur de disque dur	366	3.5	Lecture d'un nœud d'information sur disque .	401
7.5	Interruption matérielle d'un disque dur	368	4	Gestion des blocs sur nœud d'information	402
7.6	Passage des commandes	369	4.1	Détermination du numéro de bloc physique ..	402
7.7	Partitionnement des disques durs	369	4.2	Agrégation d'un bloc physique	402
7.8	Requêtes à un disque dur	370	4.3	Implémentation de la fonction auxiliaire	403
Chapitre 19 Gestion de l'antémémoire	373	5	Mise à zéro d'un nœud d'information sur disque ...	405	
1	Description des fonctions	373	5.1	Mise à zéro d'un bloc d'indirection simple ...	405
1.1	Gestion des listes de tampons	373	5.2	Mise à zéro d'un bloc d'indirection double ...	406
1.2	Fonctions d'accès aux tampons	374	5.3	Implémentation	407
1.3	Réécriture des tampons modifiés	374	6	Fonctions de service des nœuds d'information	407
2	Implémentation des fonctions de gestion de listes ..	374	6.1	Synchronisation des nœuds d'information	408
2.1	Fonctions de hachage	374	6.2	Recherche d'un nouveau descripteur de nœud d'information	408
2.2	Insertion dans les listes	374	6.3	Remplissage d'une zone de mémoire	409
2.3	Suppression des listes	375	6.4	Libération d'un nœud d'information en table des bits	410
2.4	Recherche d'un descripteur de tampon	375	6.5	Relâchement d'un nœud d'information	411
3	Réécriture sur un disque donné	376	6.6	Recherche d'un nœud d'information libre sur disque	412
4	Les fonctions de manipulation des tampons	376	6.7	Chargement d'un nœud d'information	413
4.1	Relâchement d'un tampon	376	7	Évolution du noyau	414
4.2	Détermination d'un descripteur de tampon ..	377			

Chapitre 22 Gestion interne des fichiers réguliers

et des répertoires	419
1 Montage d'un système de fichiers	419
1.1 Chargement d'un super-bloc	419
1.2 Initialisation du système de fichiers	421
1.3 Lecture de la table des partitions	422
2 Gestion des répertoires	423
2.1 Étude générale des répertoires	423
2.2 Les fichiers répertoire sous Linux	427
2.3 Fonctions internes de gestion des répertoires ..	428
3 Gestion interne des fichiers réguliers	433
3.1 Gestion des noms de fichiers	433
3.2 Lecture et écriture dans un fichier régulier ...	438
4 Évolution du noyau	440
4.1 Montage d'un système de fichiers	441
4.2 Gestion des répertoires et des fichiers	444

HUITIÈME PARTIE :

PÉRIPHÉRIQUES CARACTÈRE 449

Chapitre 23 Le clavier

1 Principe du logiciel de lecture au clavier.....	451
1.1 Modes brut et structuré	451
1.2 Tampon de lecture	451
1.3 Quelques problèmes pour le pilote	453
2 Interface du clavier sur l'IBM-PC	453
2.1 Aspect physique	453
2.2 Make-code et break-code	454
2.3 Les registres du contrôleur de clavier	454
2.4 Principe de lecture des <i>scan codes</i>	457
2.5 Le port 61h	457
3 Principe du traitement du clavier sous Linux.....	457
3.1 Le gestionnaire du clavier	458
3.2 Initialisation du gestionnaire de clavier	458
3.3 Grandes étapes du gestionnaire de clavier ...	458
4 Traitement du mode données brutes	459
4.1 Grandes étapes	459
4.2 Détermination de la fonction de traitement ..	460
4.3 Cas des touches préfixielles	461
4.4 Cas d'une touche normale	463
4.5 Les touches de déplacement du curseur	466
4.6 Les touches de fonction	467
4.7 La touche moins	468
4.8 Mise en tampon brut du clavier	468
5 Traitement du mode structuré	469
5.1 Appel	469
5.2 Passage du tampon brut au tampon structuré	470

6 Évolution du noyau	472
----------------------------	-----

Chapitre 24 Les liaisons série

1 Étude générale	477
1.1 Communication série asynchrone	477
1.2 Communication série synchrone	482
1.3 Le standard d'interface série RS-232	484
2 L'UART PC16550D	484
2.1 Le brochage	485
2.2 L'ensemble de registres	485
2.3 Programmation de l'UART	489
3 Cas de Linux	489
3.1 Initialisation des liaisons série	489
3.2 Gestionnaires d'interruption	491
4 Évolution du noyau	495

Chapitre 25 Les périphériques caractère

1 Fonctions de lecture/écriture.....	501
1.1 Fonction d'accès de haut niveau	501
1.2 Fonctions d'accès de bas niveau	501
1.3 Implémentation de la fonction d'accès de haut niveau	502
2 Fonctions d'accès de bas niveau des terminaux	503
2.1 Cas d'un terminal quelconque	503
2.2 Cas du terminal en cours	503
3 Évolution du noyau	503

NEUVIÈME PARTIE :

COMMUNICATION PAR TUBES 507

Chapitre 26 Communication par tubes sous Linux ...

1 Étude générale	509
1.1 Notion	509
1.2 Types de tubes de communication	510
2 Gestion interne sous Linux	510
2.1 Descripteur de nœud d'information d'un tube	510
2.2 Opérations d'entrée-sortie	511
3 Évolution du noyau	513

DIXIÈME PARTIE :

LE MODE UTILISATEUR 517

Chapitre 27 Appels système du système de fichiers .

1 Points de vue utilisateur et programmeur.....	519
1.1 Les fichiers du point de vue utilisateur	519
1.2 Les fichiers du point de vue du programmeur	521
2 Entrées-sorties UNIX sur fichier	522
2.1 Ouverture et fermeture de fichiers	522
2.2 Lecture et écriture de données	525

2.3	Positionnement dans un fichier	528	5.2	Implémentation	601
2.4	Sauvegarde des données modifiées	529	6	Autres appels système	603
3	Implémentation Linux des entrées-sorties	529	6.1	L'appel système break()	603
3.1	Appel système d'ouverture	529	6.2	L'appel système acct()	603
3.2	Appel système de création	533	7	Évolution du noyau	603
3.3	Appel système de fermeture	534	Chapitre 29 Les autres appels système sous Linux ...	609	
3.4	Appel système de lecture des données	534	1	Appels système de mesure du temps	609
3.5	Appel système d'écriture des données	537	1.1	Liste	609
3.6	Appel système de positionnement	538	1.2	Implémentation	610
3.7	Appel système de sauvegarde des données ...	539	2	Appels système liés à l'ordonnancement	611
4	Liens et fichiers partagés	539	2.1	Priorité des processus	611
4.1	Étude générale	539	2.2	Contrôle de l'exécution d'un processus	612
4.2	Création de liens symboliques sous UNIX ...	541	3	Appels système concernant les signaux	612
4.3	Implémentation sous Linux	542	3.1	Émission d'un signal	612
5	Manipulations des fichiers	544	3.2	Déroutement d'un signal	613
5.1	Les appels système UNIX	544	3.3	Attente d'un signal	614
5.2	Implémentation sous Linux	546	4	Appels système concernant les périphériques.....	615
6	Gestion des répertoires	550	4.1	Création d'un fichier spécial	615
6.1	Les appels système UNIX	550	4.2	Opérations de contrôle d'un périphérique ...	616
6.2	Implémentation	552	5	Appels système concernant la mémoire.....	618
7	Autres appels système	558	5.1	Structure de la mémoire utilisateur	618
7.1	Duplication de descripteur d'entrée-sortie ...	558	5.2	Changement de la taille du segment des données	619
7.2	Récupération des attributs des fichiers	559	5.3	Accès à une adresse physique	620
7.3	Dates associées aux fichiers	562	6	Tubes de communication	620
7.4	Propriétés des fichiers ouverts	563	6.1	Description	620
7.5	Montage et démontage de systèmes de fichiers	565	6.2	Implémentation	621
8	Évolution du noyau	566	7	Autres appels système	622
Chapitre 28 Appels système concernant les processus	569	8	Évolution du noyau	623	
1	Création des processus	569	Chapitre 30 Fonctions de la bibliothèque C	625	
1.1	Description des appels système	569	1	La fonction printf()	625
1.2	Implémentation de fork()	572	1.1	Description	625
1.3	Le format d'exécutable <i>a.out</i>	577	1.2	Implémentation	625
1.4	Implémentation de execve()	583	2	Fonction concernant les signaux	626
2	Gestion des attributs	592	2.1	Description	626
2.1	Description des appels système	592	2.2	Implémentation	626
2.2	Implémentation	593	3	Fonctions sur les chaînes de caractères	626
3	Gestion des groupes et des sessions de processus...	594	4	Évolution du noyau	631
3.1	Description des appels système	594	ONZIÈME PARTIE :		
3.2	Implémentation	595	DÉMARRAGE DU SYSTÈME	633	
4	Terminaison du processus en cours	596	Chapitre 31 Démarrage du système Linux	635	
4.1	Description de l'appel système	596	1	Source et grandes étapes	635
4.2	Implémentation	597	1.1	Fichiers sources concernés	635
5	Attente de la fin d'un processus fils	600	1.2	Début de l'amorçage	635
5.1	Les appels système	600	2	Le chargeur d'amorçage	636

2.1	Les grandes étapes	636	4.4	Initialisation provisoire de la table des interruptions	646
2.2	Transfert du code d'amorçage	637	4.5	Initialisation de la table globale des descripteurs	647
2.3	Configuration de la pile en mode réel	637	4.6	Valeurs finales des registres de segment de données et de pile	648
2.4	Affichage d'un message de chargement	638	4.7	Vérification de l'activation de la broche A20	649
2.5	Chargement de l'image du noyau	638	4.8	Vérification de la présence du coprocesseur arithmétique	649
3	Passage au mode protégé	642	4.9	Mise en place de la pagination	649
3.1	Les grandes étapes	642	4.10	Passage à la fonction start_kernel() ..	649
3.2	Sauvegarde de la position du curseur graphique	643	5	La fonction start_kernel()	650
3.3	Inhibition des interruptions matérielles	643	5.1	Les grandes étapes	650
3.4	Transfert du code du système	643	5.2	Initialisation du terminal	651
3.5	Chargement de tables provisoires de descripteurs	644	5.3	Passage au mode utilisateur	652
3.6	Activation de la broche A20	644	5.4	Le processus 1 : init	653
3.7	Reprogrammation du PIC	645	6	Évolution du noyau	654
3.8	Passage au mode protégé	645	Bibliographie		659
4	La fonction startup_32()	645	Index		669
4.1	Les grandes étapes	645			
4.2	Initialisation des registres de segmentation ...	646			
4.3	Configuration de la pile en mode noyau	646			