

Содержание

Предисловие	17
Введение	19
Итак...	20
Версия ядра	20
Читательская аудитория	20
Благодарности	22
Об авторе	23
От издательства	24
Глава 1. Понятие о ядре Linux	25
История систем Unix	25
Потом пришел Линус: введение в Linux	27
Обзор операционных систем и ядер	29
Отличие ядра Linux от классических ядер Unix	31
Версии ядра Linux	34
Сообщество разработчиков ядра Linux	35
Перед тем как начать	36
Глава 2. Начальные сведения о ядре Linux	37
Где взять исходный код ядра	37
Использование Git	37
Инсталляция исходного кода ядра	38
Использование заплат	38
Дерево каталогов исходных кодов ядра	39
Сборка ядра	40
Конфигурирование ядра	40
Уменьшение количества выводимых сообщений	42
Порождение нескольких параллельных задач сборки	42
Инсталляция нового ядра	42
Отличия от обычных приложений	43
Отсутствие библиотеки libc и стандартных заголовков	44
Компилятор GNU C	45
Отсутствие защиты памяти	47
Нельзя просто использовать вычисления с плавающей точкой	47
Системная стек-память небольшого фиксированного размера	47
Синхронизация и параллельное выполнение	48
Переносимость — это важно	48
Резюме	49

Глава 3. Управление процессами	51
Понятие процесса	51
Дескриптор процесса и структура <code>task_struct</code>	53
Распределение памяти под дескриптор процесса	53
Сохранение дескриптора процесса	55
Состояние процесса	56
Изменение текущего состояния процесса	58
Контекст процесса	58
Дерево семейства процессов	58
Создание нового процесса	60
Копирование при записи	60
Функция <code>fork()</code>	61
Функция <code>vfork()</code>	62
Реализация потоков в ядре Linux	63
Создание потоков	64
Потоки в ядре	65
Завершение процесса	66
Удаление дескриптора процесса	68
Дилемма “беспризорного” процесса	68
Резюме	70
Глава 4. Системный планировщик и диспетчеризация процессов	73
Мультипрограммный режим работы	73
Системный планировщик Linux	75
Стратегия планирования	76
Процессы, ориентированные на ввод-вывод и на вычисления	76
Приоритет процессов	77
Кванты времени	78
Стратегия планирования в действии	79
Алгоритм работы планировщика системы Linux	80
Классы планировщика	80
Планирование процессов в системах Unix	81
Справедливое планирование задач	83
Реализация планировщика в системе Linux	85
Учет времени	85
Выбор процесса	87
Точка входа в планировщик	91
Замораживание и активизация процессов	92
Вытеснение и переключение контекста	96
Вытеснение пространства пользователя	97
Вытеснение пространства ядра	98
Стратегии планирования в режиме реального времени	99
Системные функции для управления планировщиком	100
Системные функции для изменения стратегии и приоритета	101
Системные функции для изменения привязки к процессору	101
Передача процессорного времени другим задачам	102
Резюме	102

8 Содержание

Глава 5. Системные функции	103
Взаимодействие с ядром	103
API, POSIX и библиотека C	104
Системные функции	105
Номера системных функций	106
Быстродействие системных функций	107
Обработчик вызова системных функций	107
Как определить, какую системную функцию вызвать	108
Передача параметров	109
Реализация системных функций	109
Разработка системных функций	109
Проверка параметров	110
Контекст системной функции	113
Завершающие этапы регистрации системной функции	114
Доступ к системным функциям из пользовательских приложений	116
Почему не нужно создавать системные функции	117
Резюме	118
Глава 6. Структуры данных ядра	119
Связанные списки	119
Однонаправленный и двунаправленный связанный список	120
Циклически связанные списки	120
Перемещение по элементам связанного списка	121
Реализация в ядре Linux	122
Работа со связанными списками	124
Обход элементов связанного списка	127
Очереди	130
Система kfifo	131
Создание очереди	132
Постановка в очередь	132
Выборка из очереди	132
Определение размера очереди	133
Очистка и удаление очереди	133
Примеры использования очередей	134
Таблицы отображения	134
Инициализация структуры idr	135
Выделение нового UID	135
Поиск UID	137
Удаление UID	137
Аннулирование idr	137
Двоичные деревья	138
Двоичные деревья поиска	138
Самобалансирующиеся двоичные деревья поиска	139
Красно-черные деревья	139
Реализация в Linux	140
Какие структуры данных следует использовать, если...	142
Алгоритмическая сложность	143
Что такое алгоритм?	143

Понятие большого “О”	144
Понятие большой “тета”	144
Временная сложность алгоритма	145
Резюме	146
Глава 7. Прерывания и их обработка	147
Прерывания	147
Обработчики прерываний	149
Верхняя и нижняя половины	150
Регистрация обработчика прерывания	150
Флаги обработчика прерываний	151
Остальные параметры функции обработки прерывания	152
Пример обработчика прерывания	153
Освобождение обработчика прерывания	153
Написание обработчика прерывания	154
Обработчики общих запросов на прерывание	155
Пример настоящего обработчика прерывания	156
Контекст прерывания	158
Реализация системы обработки прерываний	159
Интерфейс /proc/interrupts	162
Управление прерываниями	163
Запрещение и разрешение прерываний	164
Запрещение заданной линии IRQ	165
Состояние системы обработки прерываний	166
Резюме	167
Глава 8. Нижняя половина обработчика и отложенные действия	169
Нижняя половина	170
Когда используется нижняя половина обработчика	171
Многообразие нижних половин	171
Отложенные прерывания	174
Реализация механизма отложенных прерываний	175
Использование отложенных прерываний	177
Тасклеты	179
Реализация тасклетов	179
Использование тасклетов	182
Демон ksoftirqd	184
Старый механизм ВН	186
Очереди отложенных действий	187
Реализация очередей отложенных действий	188
Использование очередей отложенных действий	191
Старый механизм очередей задач	194
Какие механизмы обработчиков нижних половин следует использовать	195
Блокировки между нижними половинами обработчиков	196
Запрещение обработки нижних половин	197
Резюме	199

10 Содержание

Глава 9. Общие сведения о синхронизации кода ядра	201
Критические участки и конфликты из-за доступа к системным ресурсам	202
Зачем вообще нужно что-то защищать?	202
Общая переменная	204
Блокировки	205
Причины возникновения параллелизма	207
Что нужно защищать?	209
Взаимоблокировки	210
Конфликт при блокировке и масштабируемость	213
Резюме	215
Глава 10. Средства синхронизации ядра	217
Неделимые операции	217
Неделимые целочисленные операции	218
64-разрядные неделимые операции	222
Неделимые битовые операции	223
Спин-блокировки	225
Функции для спин-блокировки	227
Другие средства работы со спин-блокировками	229
Спин-блокировки и нижние половины обработчиков прерываний	230
Спин-блокировки по чтению-записи	230
Семафоры	233
Счетные и бинарные семафоры	234
Создание и инициализация семафоров	235
Использование семафоров	236
Семафоры для чтения-записи	237
Мьютексы	238
Сравнение семафоров и мьютексов	240
Сравнение спин-блокировок и мьютексов	240
Условные переменные	241
ВКЛ: большая блокировка ядра	242
Последовательные блокировки	243
Отключение мультипрограммного режима работы ядра	245
Порядок выполнения операций и барьеры	247
Резюме	251
Глава 11. Таймеры и управление временем	253
Основная идея учета времени в ядре	254
Частота импульсов таймера: директива HZ	255
Идеальное значение параметра HZ	256
Преимущества больших значений параметра HZ	257
Недостатки больших значений параметра HZ	258
Переменная jiffies	259
Внутреннее представление переменной jiffies	260
Переполнение переменной jiffies	261
Пользовательские программы и параметр HZ	263
Аппаратные часы и таймеры	264
Часы реального времени	264

Системный таймер	264
Обработчик прерываний от таймера	265
Абсолютное время	267
Таймеры	269
Использование таймеров	270
Конфликты из-за доступа к ресурсам при использовании таймеров	272
Реализация таймеров	272
Задержка выполнения	273
Задержка с помощью цикла	273
Короткие задержки	274
Функция <code>schedule_timeout()</code>	276
Резюме	278
Глава 12. Управление памятью	279
Страничная организация памяти	279
Зоны	281
Выделение страниц памяти	284
Выделение обнуленных страниц памяти	284
Освобождение страниц памяти	285
Функция <code>kmalloc()</code>	286
Флаги <code>gfp_mask</code>	287
Функция <code>kfree()</code>	292
Функция <code>vmalloc()</code>	292
Уровень блочного распределения памяти	294
Структура уровня блочного распределения памяти	295
Интерфейс блочного распределителя памяти	298
Пример использования блочного распределителя памяти	300
Статическое выделение памяти в стеке	301
Одностраничные стеки ядра	302
Справедливое использование стека	303
Отображение верхней памяти	303
Постоянное отображение	303
Временное отображение	304
Выделение памяти для конкретного процессора	305
Новый интерфейс <code>regsru</code>	306
Работа с процессорными данными на этапе компиляции	306
Работа с процессорными данными на этапе выполнения	307
Когда лучше использовать данные, связанные с процессорами	308
Выбор способа выделения памяти	309
Резюме	310
Глава 13. Виртуальная файловая система	311
Общий интерфейс файловых систем	312
Абстрактный уровень файловой системы	312
Файловые системы Unix	314
Объекты VFS и их структуры данных	315
Объект суперблок	317
Операции суперблока	318

12 Содержание

Объект inode	320
Операции с файловыми индексами	322
Объект элемента каталога (dentry)	325
Состояние элементов каталога	326
Кеш объектов dentry	327
Операции с элементами каталогов	328
Файловый объект	329
Файловые операции	330
Структуры данных, связанные с файловыми системами	335
Структуры данных, связанные с процессом	337
Резюме	339

Глава 14. Уровень блочного ввода-вывода 341

Структура блочного устройства	342
Буферы и их заголовки	343
Структура bio	346
Векторы ввода-вывода	347
Сравнение старой и новой реализаций	348
Очереди запросов	349
Планировщики ввода-вывода	350
Задачи планировщика ввода-вывода	350
Лифт имени Линуса	351
Планировщик ввода-вывода с ограничением по времени	353
Прогнозирующий планировщик ввода-вывода	355
Планировщик ввода-вывода с полностью равноправными очередями	356
Планировщик ввода-вывода с отсутствием операций (Noop)	357
Выбор планировщика ввода-вывода	357
Резюме	358

Глава 15. Адресное пространство процесса 359

Адресные пространства	359
Дескриптор памяти	361
Выделение дескриптора памяти	363
Удаление дескриптора памяти	363
Структура mm_struct и потоки ядра	364
Области виртуальной памяти	364
Флаги областей VMA	365
Операции с областями VMA	367
Списки и деревья областей памяти	368
Области памяти в реальных приложениях	369
Работа с областями памяти	371
Функция find_vma()	371
Функция find_vma_prev()	372
Функция find_VMA_intersection()	372
Функции mmap() и do_mmap(): создание диапазона адресов	373
Функции munmap() и do_munmap(): удаление диапазона адресов	375
Таблицы страниц	375
Резюме	377

Глава 16. Страничный кеш и отложенная запись страниц	379
Методики кеширования	380
Кеширование при записи	380
Вытеснение данных из кеша	381
Реализация страничного кеша в ОС Linux	383
Объект address_space	383
Операции объекта address_space	385
Базисное дерево	387
Старая хеш-таблица страниц	387
Буферный кеш	388
Потоки синхронизатора	388
Режим ноутбука	390
Экскурс в историю: bdflush, kupdated и pdflush	391
Предотвращение перегрузки с помощью нескольких потоков	392
Резюме	393
Глава 17. Устройства и модули	395
Типы устройств	395
Модули	396
Модуль “Hello, World!”	397
Сборка модулей	398
Установка модулей	401
Генерация зависимостей между модулями	401
Загрузка модулей	401
Поддержка параметров конфигурации	402
Параметры модулей	404
Экспортируемые символы	406
Модель представления устройств	407
Объекты kobject	408
Типы ktype	409
Множества объектов kset	410
Взаимосвязь kobject, ktype и kset	410
Управление и работа с объектами kobject	411
Счетчики ссылок	412
Файловая система sysfs	414
Добавление и удаление объектов файловой системы sysfs	417
Добавление файлов в файловую систему sysfs	418
Уровень событий ядра	421
Резюме	423
Глава 18. Отладка	425
Начало работы	425
Ошибки ядра	426
Отладка с помощью вывода диагностических сообщений	427
Устойчивость	427
Уровни вывода сообщений ядра	428
Буфер сообщений ядра	429
Демоны syslogd и klogd	429

14 Содержание

Взаимозаменяемость функций printf() и printk()	430
Сообщения Oops	430
Утилита ksymoops	431
Функция kallsyms	432
Параметры конфигурации для отладки ядра	433
Объявление об ошибках и выдача информации	433
“Магическая” клавиша <SysRq>	434
Сага об отладчике ядра	435
Отладчик gdb	436
Отладчик kgdb	436
Исследование и тестирование системы	437
Использование идентификатора UID в качестве условия	437
Использование условных переменных	437
Использование статистики	438
Ограничение частоты следования и общего количества событий при отладке	438
Поиск методом половинного деления изменений, приводящим к ошибкам	439
Поиск с помощью git	440
Если ничто не помогает — обратитесь к сообществу	441
Резюме	441
Глава 19. Переносимость	443
Переносимые операционные системы	443
История переносимости Linux	445
Размер машинного слова и типы данных	446
Скрытые типы данных	449
Специальные типы данных	449
Типы с явным указанием размера	450
Знаковые и беззнаковые типы char	451
Выравнивание данных	451
Как избежать проблем с выравниванием	452
Выравнивание нестандартных типов данных	452
Пустые поля структур	453
Порядок следования байтов	454
Учет времени	456
Размер страницы памяти	457
Порядок выполнения операций процессором	458
Многопроцессорность, мультипрограммирование и верхняя память	458
Резюме	459
Глава 20. Заплаты, хакерство и сообщество	461
Сообщество	461
Стиль написания исходного кода	462
Отступы	462
Оператор switch	463
Пробелы	463
Фигурные скобки	464
Длина строки исходного кода	465
Соглашения о присвоении имен	466

Содержание 15

Функции	466
Комментарии	466
Использование директивы typedef	467
Использование того, что уже есть	468
Избегайте директив ifdef в исходном коде	468
Инициализация структур	468
Исправление ранее написанного кода	469
Организация команды разработчиков	469
Отправка сообщений об ошибках	470
Заплаты	470
Генерация заплат	470
Генерирование заплат с помощью программы Git	471
Публикация заплат	472
Резюме	473
Список литературы	475
Книги по основам построения операционных систем	475
Книги о ядре Unix	476
Книги о ядре Linux	476
Книги о ядрах других операционных систем	476
Книги по API Unix	477
Книги по программированию на языке C	477
Другие работы	477
Веб-сайты	478
Предметный указатель	479