1 лаб. работа – написание демона и написание программы вывода на экран дерева каталога, которая выводит на экран. Стивин Раго весь код можно взять оттуда.

7 типов файлов:

* обычные
* директории
* хардлинки
* Пайпы
* сокеты
* специальный символьный файл
* специальный блочный файл

Рекурсию при выводе дерева каталога попробовать переписать на итерацию. Найти условие выхода из рекурсии.

В коде убираем Go to и continue.

Windows, Unix, Linux – системы с монолитными ядрами.

Windows включает в себя многоуровневые драйвера.

TaskStruct

Виртуальная файловая система /proc – не находится на диске, создается на лету. Фартунки copy to user.

Буферизованный, не буферизованный ввод вывод.

Системный вызов /open. Выполняется по мануалу, рассматриваются процессы происходящие внутри функции open.

Пишем свою виртуальную файловую систему.

Программы связанные с прерванием. Системы связанные с прерыванием делятся на 2 части. Tok haf and Botom haf. Tasklet и очередь работ

5 моделей ввода вывода с точки зрения программиста.

Сокеты и мультиплексирование. Пишем клиент-сервер, для двух типов сокетов.

Драйвер имеет множество точек входа. Системные вызовы, аппаратные прерывания являются точками входа для ОС.

Глава 13 Стивина Рага демоны. Листнинг 13.3. Вызывается функция Demonise. Данная функция преобразует процесс в демона и реализуется 6ю правилами. Далее вызывается функция All ready running. –ps. Демона можно удалить командой kill.

#ps-ajx – вывод процессов, которыми владеют другие пользователи

-а- вывод процессов не имеющих управляющего терминала

-х- вывод дополнительные сведения имеющие отношения к заданию, а именно идентификатор сессии, идентификатор группы и идентификатор группы процессов терминала.

6 правил программирования демонов:

* вызывается функция umask(0). Эта функция сбрасывает маску создания файлов. У Стивина данная функция вызывается из кода предка и потом передается по наследству потомку при втором шаге. Также демон сможет создавать файлы с любыми правами доступа.
* вызывается системный вызов fork(). После создания потомка, процесс предка завершается. Почерний процесс не является лидером группы.
* вызывается функция setsid(). Делает демона лидером сессии. У демона нет управляющего терминала.
* chdir(‘/’) переходим в корень, т.к. демон может быть запущен с флешки.
* Закрыть все ненужные файловые дескрипторы и оставляем только 3.
* Отправляем файлы на /dev/null

Литература:

Саламон Русинович

Олег Церюлик взаимодействие модулей ядра.

Стивин Раго

Рихтар

Вахалия