ОС. Семинар 4

05.10.2024

К лабе 2 unix: сигналы

⑤ Стивен Раго, пример 2 демон.

Системный вызов exec()

- переводит процесс на новое адр. простр.-во (адр. пр-во программы, передаваемой в кач. параметра).

Также говорят, что с.в. exec создаёт низкоур. процесс.

Что это значит? exec не создаёт новый процесс, ↳ создаёт АП.

См. ↓

- (Из методички к лаб 2) Шаги выполнения:

Разбирает путь к исп. файлу.

! Для системы сущ-ет только полное имя файла (нач. с корн. кат. - "/"). - строка, которую разбирает по элементам пути (dentry) до каждого след. "/" (чтобы система определила, существует ли файл.), где каждый эл-т пути - каталог.

Нет полноценной выполн. потоки ядра.

В Unix расширения не нужны, т.к. любая программа имеет заголовок.

Выполняется перечень проверок:

- сущ. файла
- исполн.
- права доступа (x)

- сист. вызов exec() - созд. АП.

! Режимы: user
kernel } - два разных режима

В ядро невозможно напрямую передавать, для этого существует спец. функции ядра: могут вызываться последовательно много функций, ветвление выполняется по флагам.

! В системе все флаги предопределены. Это же касается ошибок.

Имя ф-ций - строка. switch ?

(?)

Область свопинга, пейджинг - страница на стр-цу.

Таблица страниц расп-ся в обл. данных ядра. - крайне дефицит. (ресурс) АП. Данные не удаляются/высвобождаются, а помечаются, как свободные, после чего необходимо выполнить замещение.

! Поменять одно АП на другое - просто поменять указатель (см. struct proc, struct task)

! vfork() - новое АП не создаётся: предок блок., а потомок работает в его АП.
Потомок переходит на АП, созд. exec'ом.

Иниц. апп. контекста: В проц. INTEL есть регистр **CR3** - в этом регистр необходимо поместить адр. таблицы страниц для начала преобраз. адр-в вирт. к физич.

В арх x86-64 адр. таблица $4^{го}$ уровня.
В СК3 - режим long.

! Усыновление связано с изм. указателей у parent/child. (введён wait())

Для единообразия система рассм. проц, как поток.