ЖЦ Процесса

Появление процесса в системе

Init – первый процесс в системе(все чаще systemd)

Функция fork() порождает новый процесс со всеми его атрибутами и ресурсами (сист.вызов clone())

После вызова fork() дочерний родительский процесс продолжают работу с точки вызова fork()

Дочернему процессу fork() возвращает 0, родительскому – PiD порождение процесса.

Дочерний процесс – полная копия родительского, за исключением PID.

Первым получит управление child.

Copy-on-write

Адресное пространство родителя и порождение процесса совместно

Дубликат ресурсов создается только при попытке изменить данные

Для создания потока используется тот же системный вызов clone()

Как появляется новый процесс : exec

Exec загружает бинарный файл в адресное пространство процесса

Выполнение начинается с точки входа в программу

При выполнение сразу после fork() копирование страниц памяти не происходит

Смерть процеса

Процесс заканчивается системным вызовов exit() либо сигналом завершение

Сигнал процессу можно отправить командой kill

После завершение дочерний процесс находится в состояние зомби

Если родительский процесс завершится раньше дочернего – родительским становится либо другой потом родительского процесса, либо init (pid = 1)

Статус завершения интерактивного процесса доступен в переменной окружения $? – она хранит код завершение последней команд.

Состояние процесса : Ready

Ready – готов к выполнению

Ждет выделения процессерного времени планировщиком

Посмотреть очередь процессов команда – load average

Состояние процесса : Running

Как только планировщик поставил процесс на выполнение, процесс переходит в состояние running

Если процесс в состоянии Stopedd(T) то помочь ему вернутся в running можно сигналом SIGCONT

Если мы отправили его спать с ctrl + z то команда fg, bg выведут его на выполнение

Состояние процесса : остановлен

Остановлен - прекращает работу до получения сигнала signcont(?)

Или signkill(?)

Используют трассировщики

Сочетание клавиш ctrl+z или сигнал sigstop отправят процесс в это состояние

Трассировщики - это программы которые запускают коды пошагово, они используется для откладки (в vs это брейкпоинты)

Состояние процесса : Sleeping

Sleeping(s) – процесс в ожидании, пока запрашиваемый ресурс станет доступен

Вызов функции sleep() отправляет процесс в состояние s

Как только ресурс доступен процесс переходит в состояние ready или running uninterruptable sleep(D) то же, что и выше, но процесс не обрабатывает поступающие сигналы до завершения системного вызова

Состояние процесса : Zombie

Зомби (Z) - процесс завершился, но его родитель еще не обработал waitpid()

В состоянии zombie процесс уже не занимает системные ресурсы, только дескриптор процесса (чтобы сообщить родителю статус завершение )



