**Планировщик EDF**

**Руководство пользователя**

Санкт - Петербург

2017

**Введение**

Политика планирования для задачи уровня пользователя может быть установлена ​​в SCHED\_DEADLINE для достижения двух разных результатов:

* Гарантии времени: задача гарантирована определенной долей времени процессора, независимо от поведения других задач, выполняемых в системе; Эти гарантии важны для приложений, чувствительных к времени (например, управления в реальном времени), которые должны выполнять определенную работу в рамках ограничения времени.
* Ограничение сроков: задача вынуждена использовать не более, чем назначенный ей ресурс ЦП; Это полезно для ограничения количества CPU, используемого высокоприоритетными задачами, которые в случае ошибки или интенсивного вычисления могут голодать остальную часть системы. Это также полезно в средах виртуализации для справедливого распределения ЦП между несколькими виртуальными машинами.

В обоих случаях свойство временной изоляции используется. В первом случае это изоляция от других задач, в интересах самой задачи. В последнем случае вместо этого это изоляция для других задач.

**Подготовка к работе**

1. С помощью утилиты wget загружаем исходный код ядра:

sudo wget https://www.kernel.org/pub/linux/kernel/v2.6/linux-2.6.24.tar.bz2

1. Распаковываем архив:

tar –xjvf linux-2.6.24.tar.bz2

1. Создаем конфигурационный файл из текущего системного конфигурационного файла:

sudo cp /boot/config-2.6.24 .config

1. Меняем параметр extraversion в Makefile, чтобы отличать собираемое ядро от других версий

EXTRAVERSION= -edf

1. Компилируем ядро:

sudo make oldconfig

sudo make-kpkg –initrd kernel\_image 2>../errors

1. Начинается компиляция ядра, и если все идет хорошо, создается сжатый образ ядра, в противном случае можно проверить ошибки в файле ошибки, созданном в домашнем каталоге
2. Установим скомпилированную версию ядра, сгенерированную предыдущим шагом

sudo dpkg -i kernel\_image-2.6.24\_xxxx.deb

1. Для завершения процесса необходимо перезагрузить систему и выбрать загрузку нового ядра:



Рис.4. Загрузка с новой версии ядра

**Использование**

Для изменения политики планирования процесса используется системный вызов sched\_setscheduler. Системный вызов sched\_setscheduler устанавливает как политику планирования, так и соответствующие параметры для процесса, идентифицируемого параметром pid. Если pid равен 0, устанавливаются параметры планировщика, применяемые к вызвавшему процессу.

Соответствующая служебная процедура sys\_sched\_setscheduler просто вызывает функцию do\_sched\_setscheduler . Эта функция проверяет допустимость политики планирования, определяемой параметром policy, и нового приоритета, определяемого параметром param->sched\_priority. Она также проверяет, есть ли у процесса способность cap\_sys\_nice, или наличие прав суперпользователя у его владельца. Если все в порядке, она удаляет процесс из очереди на выполнение (если он выполняемый), обновляет статический и динамический приоритеты и приоритет реального времени у процесса, возвращает процесс в очередь на выполнение и, если необходимо, вызывает функцию resched\_task для вытеснения текущего процесса, принадлежащего данной очереди.