**Лабораторная работа №12.**

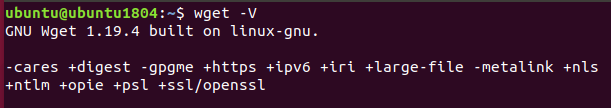
**Сборка ядра Linux.**

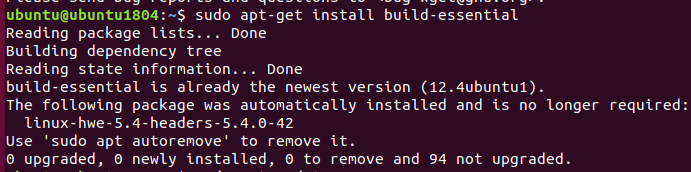
Цель лабораторной работы – научиться выполнять сборку ядра Linux.

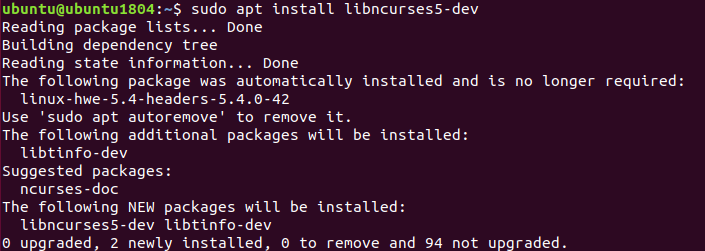
Зачем пересобирать ядро? Обычно этого делать не нужно, так как ядро, устанавливаемое по умолчанию в Debian, подходит для многих конфигураций. Кроме этого, в Debian есть ещё несколько сборок ядра. Поэтому, сначала проверьте, может другой пакет с ядром лучше подходит для вашего компьютера. Однако, полезно пересобрать ядро в следующих случаях:

* если установлено специфичное оборудование или возникает конфликт аппаратного обеспечения со стандартным ядром
* чтобы задействовать свойства, которых нет в поставляемых сборках ядра (например, поддержка high memory)
* для оптимизации ядра, удаляя ненужные драйверы для уменьшения времени загрузки
* создания монолитного ядра, без модулей
* для установки обновлённого или разрабатываемого ядра
* чтобы больше узнать о ядрах linux

**Задание 1.** Выясните назначение пакетов *wget*, *build-essential*, *libncurses5-dev* и установить их.







**Задание 2.** Узнайте версию своего ядра можно командой:  
 *uname –a*  
**Задание 3.** Скачайте в папку, количество свободного места в которой которой не меньше 1 ГБ, другую стабильную версию ядра с сайта [www.kernel.org](http://www.kernel.org/), например:

*wget* [*http://cdn.kernel.org/pub/linux/kernel/v4.x/linux-4.8.6.tar.xz*](http://cdn.kernel.org/pub/linux/kernel/v4.x/linux-4.8.6.tar.xz)

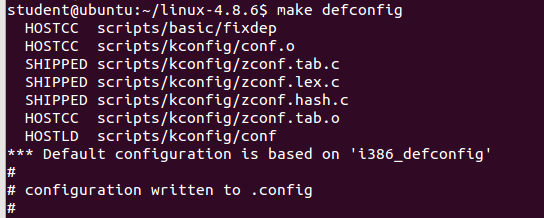
**Задание 4.** Распакуйте архив, например:

*tar xf linux-4.8.6.tar.xz*



**Задание 5.** Начинаем конфигурирование опций ядра. Зайдите в разархивированную папку и выполните команду:

*make defconfig*



**Задание 6.** Выясните, для чего используется defconfig.



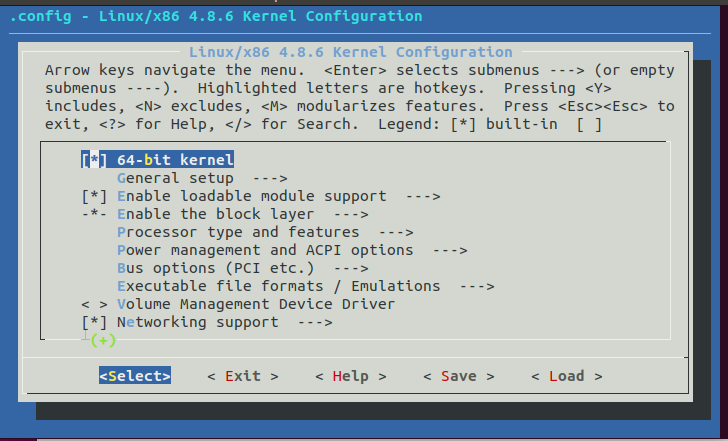
Новая конфигурация по умолчанию из файла defconfig, предоставленного ARCH

**Задание 7.** Что произойдет, если выполнить команду?

*make menuconfig*



*обновите текущую конфигурацию с помощью программы на основе меню*



**Задание 8.** Выполните сборку ядра:

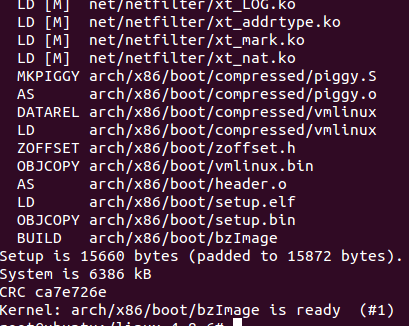
make –j 5

За что отвечает параметр –j?

Определяет число работ - jobs (commands) которые запускаются одновременно. Если указано больше чем одна -**j** опция, то применяться будет последняя. Если -**j** опция дана без аргументов **make** не будет ограничивать количество работ запускаемых одновременно. То есть запуститься сможет неограниченное количество работ одновременно.

Посмотрите размер и местоположение ядра, код для проверки.

Успешная проверка завершается следующим образом:



**Задание 9.** Загрузите ядро в систему, убедитесь, что после перезагрузки операционной системы появилась возможность выбора ядра для загрузки.

