Embedded Linux course - Kernel

Total des points 3/10

✓ What's the license of the Linux kernel? * 1/1

- (•) GNU GPLv2 ✓
- ( ) GNU GPLv3

Commentaire

GNU GPLv2 is the correct answer. You have no obligation to allow users to run modified versions of the GPL software on the device.

✗ Version 5.0 marks a noticeable milestone in Linux kernel development * 0/1

- (•) True ✗
- ( ) False

Bonne réponse

- (•) False

Commentaire

Wrong. The number of changes between 4.20 and 5.0 is not substancially different from the number of changes between 4.19 and 4.20. Increasing the version number to 5.0 was just a way to highlight the amount of change that happened since 4.0, and to make 4.x kernels look older.

✓ Who manages the Linux merge Window? * 1/1

- (●) Linus Torvalds ✓
- ( ) Greg Kroah Hartmann
- ( ) Andrew Morton

Commentaire

After a release is made, Torvalds, being the kernel maintainer, is ready to accept ("merge") changes for the next version during the "merge window". Once he closes the window, he will generally only accept bug fixes, but no new fixes.

✓ If your hardware is fully supported and no feature is missing, it's your best interest to use: * 1/1

- (●) Mainline Linux from https://kernel.org ✓
- ( ) Linux sources published by your System On Chip provider

Commentaire

Correct, at least in Bootlin's opinion, mainline Linux is best because it offers the guarantee that your hardware will also be supported in the next kernel releases and that community support will be available too. With the vendor kernel, this probably won't happen.

✗ To upgrade from Linux 5.6.10 to 5.6.11, that's sufficient to apply the patch-5.6.11.xz patch * 0/1

- (●) True ✗
- ( ) False

Bonne réponse

- (●) False

Commentaire

Wrong. patch-5.6.11 contains all the changes between 5.6 and 5.6.11. Applying this patch on 5.6.10 would try to re-apply all the patches from 5.6 and 5.6.10.
To apply the 5.6.11 patch, you first have to revert to the original 5.6 release (applying the 5.6.10 patch backwards)

✗ defconfig files contain * 0/1

- (●) Default configuration settings for a given board or CPU family ✗
- ( ) Non default configuration settings for a given board or CPU family

Bonne réponse

- (●) Non default configuration settings for a given board or CPU family

Commentaire

Wrong. defconfig files precisely only contain settings which value is different from its default value. This way, only meaningful kernel parameters are stored in such files.

✗ Modprobe can be used to remove kernel modules * 0/1

- ○ True
- ◉ False ✗

Bonne réponse

- ◉ True

Commentaire

Wrong. modprobe has a "-r" option to remove a module and its no longer needed dependencies.

✗ You can compile a kernel supporting two different ARM SoC families at the same time * 0/1

- ○ True
- ◉ False ✗

Bonne réponse

- ◉ True

Commentaire

Wrong. A binary kernel can support many different boards with different ARM CPUs at the same type (only 32 bit or 64 bit at the same time). Thanks to the device tree passed by the bootloader, the kernel knows what SOCs and devices are present and therefore should be initialized.

✗ Kernel modules are mostly used in * 0/1

- ○ Desktop and server Linux systems
- ◉ Embedded Linux systems ✗

Bonne réponse

- ◉ Desktop and server Linux systems

Commentaire

Wrong. Kernel modules are mostly used in server and desktop systems, to reduce the size of the kernel that can support many different hardware devices and configurations. In embedded systems, especially dedicated ones with fixed hardware, you can easily do without kernel modules. Kernel modules are mostly useful for reducing boot time.

✗ From the root user, you can modify module parameter values after module loading * 0/1

- ◉ Always true ✗
- ○ Not always true

Bonne réponse

- ◉ Not always true

Commentaire

Wrong. It's only possible to modify module parameter values after module loading if that's allowed in the module source code.