Upgrading Debian kernel (5.4 to 5.10)

I have recently upgraded one of my Debian servers from 10 to 11. The main issue I ran into was the fact that the kernel was backported. This meant that when running Debian 10, the kernel was newer than original (5.4 vs 4.19), but now that I have upgraded to Debian 11, which ships with 5.10 by default, I am now running older version than default. This is nothing to worry about, since the 5.4 is the 20th LTS (Long-Term-Support) release, meaning it will be supported up until December 2025. The sole reason why am I even doing this is to have the default and newest Debian kernel and avoid having a FrankenDebian.

Check current version

First, check the current version of your kernel.

```
$ uname -r


5.4.0-74-generic
```

Update and upgrade

It is always recommended to update apt repositories and upgrade any pending packages.

```
$ sudo apt update
$ sudo apt upgrade
```

Search apt

Issue a search query to apt to see which kernel versions are available in the default repository.

```
$ apt-cache search linux-image | grep amd64


linux-headers-5.10.0-8-amd64 - Header files for Linux 5.10.0-8-amd64
linux-headers-5.10.0-8-cloud-amd64 - Header files for Linux 5.10.0-8-cloud-amd64
linux-headers-5.10.0-8-rt-amd64 - Header files for Linux 5.10.0-8-rt-amd64
linux-image-5.10.0-8-amd64-dbg - Debug symbols for linux-image-5.10.0-8-amd64
linux-image-5.10.0-8-amd64-unsigned - Linux 5.10 for 64-bit PCs
```

```
linux-image-5.10.0-8-cloud-amd64-dbg - Debug symbols for linux-image-5.10.0-8-cloud-amd64
linux-image-5.10.0-8-cloud-amd64-unsigned - Linux 5.10 for x86-64 cloud
linux-image-5.10.0-8-rt-amd64-dbg - Debug symbols for linux-image-5.10.0-8-rt-amd64
linux-image-5.10.0-8-rt-amd64-unsigned - Linux 5.10 for 64-bit PCs, PREEMPT_RT
linux-image-amd64-dbg - Debugging symbols for Linux amd64 configuration (meta-package)
linux-image-amd64-signed-template - Template for signed linux-image packages for amd64
linux-image-cloud-amd64-dbg - Debugging symbols for Linux cloud-amd64 configuration (meta-package)
linux-image-rt-amd64-dbg - Debugging symbols for Linux rt-amd64 configuration (meta-package)
linux-image-5.10.0-8-amd64 - Linux 5.10 for 64-bit PCs (signed)
linux-image-5.10.0-8-cloud-amd64 - Linux 5.10 for x86-64 cloud (signed)
linux-image-5.10.0-8-rt-amd64 - Linux 5.10 for 64-bit PCs, PREEMPT_RT (signed)
linux-image-amd64 - Linux for 64-bit PCs (meta-package)
linux-image-cloud-amd64 - Linux for x86-64 cloud (meta-package)
linux-image-rt-amd64 - Linux for 64-bit PCs (meta-package)
```

Pick the right image and headers for you. In my case it is the `linux-image-5.10.0-8-amd64` and `linux-headers-5.10.0-8-amd64`

Install new kernel packages

```
$ sudo apt install linux-image-5.10.0-8-amd64 linux-headers-5.10.0-8-amd64
```

In my case the system wanted/needed to install some additional packages as well, don't worry about that too much right now, we can check what was installed later and possibly uninstall if we want to.

The last lines of the installation should look like this:

```
Setting up linux-headers-5.10.0-8-amd64 (5.10.46-4) ...
Setting up initramfs-tools-core (0.140) ...
Setting up initramfs-tools (0.140) ...
update-initramfs: deferring update (trigger activated)
Setting up linux-image-5.10.0-8-amd64 (5.10.46-4) ...
I: /vmlinuz.old is now a symlink to boot/vmlinuz-5.10.0-8-amd64
I: /initrd.img.old is now a symlink to boot/initrd.img-5.10.0-8-amd64
I: /vmlinuz is now a symlink to boot/vmlinuz-5.10.0-8-amd64
I: /initrd.img is now a symlink to boot/initrd.img-5.10.0-8-amd64
/etc/kernel/postinst.d/initramfs-tools:
update-initramfs: Generating /boot/initrd.img-5.10.0-8-amd64
Processing triggers for initramfs-tools (0.140) ...
update-initramfs: Generating /boot/initrd.img-5.10.0-8-amd64
```

Reboot to apply changes

Such a big change as upgrading the kernel requires the system to be rebooted. Until then, the system will use the old kernel. Even in case the new kernel doesn't boot, you can still use the old one, because the new was just installed alongside with the old.

```
$ sudo reboot
```

> It is recommended to have access to the booloader (recovery) in case something goes wrong. **Procceed with caution if you have SSH access only.**

> You system should now reboot with the newest kernel. In my case, the VPS booted back to the old kernel, probably because the provider has hard set it to boot only that specific kernel. Consult your provider in case you have a VPS.

Revision #3
Created 15 September 2021 21:48:56 by Marek
Updated 22 September 2021 21:56:14 by Marek