How to switch from the Huge to the Generic kernel?

At the end of this recipe the user will be able to switch from the Huge to the Generic kernel on Slackware, and enjoy its benefits. You may be wondering what benefit you get from this change. The benefits of switching from Huge to Generic are:

- More fluidity and faster boot, due to Generic loading only the modules your machine needs.
- The modules are not embedded, thus giving better flexibility.

So what is the Huge kernel for? Because it has 'huge' support for many drivers built right into the kernel file, it is ideal for use in installing on a wide range of hardware. After installation a Huge kernel is no longer needed, which is why we should make the switch.

Starting the recipe

Login as root, every step in the process will be done by the superuser. After login we will execute the command '`mkinitrd_command_generator.sh` which is present in */usr/share/mkinitrd*

```
/usr/share/mkinitrd/mkinitrd_command_generator.sh
```

The output will be an example`mkinitrd` command-line which containsall the modules necessary for booting your machine hardware and accessing the root filesystem. Example output:

```
/usr/share/mkinitrd/mkinitrd_command_generator.sh
#
# mkinitrd_command_generator.sh revision 1.45
#
# This script will now make a recommendation about the command to use
# in case you require an initrd image to boot a kernel that does not
# have support for your storage or root filesystem built in
# (such as the Slackware 'generic' kernels').
# A suitable 'mkinitrd' command will be:

mkinitrd -c -k 4.4.172 -f ext4 -r /dev/sda1 -m usb-storage:ehci-hcd:ehci-pci:xhci-pci:ohci-pci:xhci-hcd:uhci-hcd:hid:usbhid:i2c-hid:hid_generic:hid-cherry:hid-logitech:hid-logitech-dj:hid-logitech-hidpp:hid-lenovo:hid-microsoft:hid_multitouch:jbd2:mbcache:ext4 -u -o /boot/initrd.gz
```

Copy the entire line where it starts with the command '`mkinitrd -c -k …`', paste it into your terminal and press ENTER. The `mkinitrd` command will be executed and will have output in this format:

```
mkinitrd -c -k 4.4.172 -f ext4 -r /dev/sda1 -m usb-storage:ehci-hcd:ehci-pci:xhci-pci:ohci-pci:xhci-hcd:uhci-hcd:hid:usbhid:i2c-hid:hid_generic:hid-cherry:hid-logitech:hid-logitech-dj:hid-logitech-hidpp:hid-lenovo:hid-
```

```
microsoft:hid_multitouch:jbd2:mbcache:ext4 -u -o /boot/initrd.gz

OK: /lib/modules/4.4.172/kernel/drivers/usb/storage/usb-storage.ko added.
OK: /lib/modules/4.4.172/kernel/drivers/usb/host/ehci-hcd.ko added.
OK: /lib/modules/4.4.172/kernel/drivers/usb/host/ehci-hcd.ko added.
OK: /lib/modules/4.4.172/kernel/drivers/usb/host/ehci-pci.ko added.
...
38432 blocks
/boot/initrd.gz created.
Be sure to run lilo again if you use it.
```

Tip: the following command will automatically execute the `mkinitrd` commandline so that you do not have to copy and paste it yourself:

```
$(/usr/share/mkinitrd/mkinitrd_command_generator.sh -r)
```

This is useful to execute the command within a script where you can also pass additional parameters such as the partition UUID and a specific file name for initrd.gz:

```
$(/usr/share/mkinitrd/mkinitrd_command_generator.sh --run
/boot/vmlinuz-generic-4.4.172 -a "-o
/boot/initrd_specific_file_name.gz -r "UUID=25a4dafe-bbf2-413f-
a60c-8c38efc0a122"")
```

In the note above replace the kernel number, the UUID 32 digits hexadecimal number and the initrd filename based on what you have on you system.

After that we will enter in lilo the information needed for Kernel Generic to be able to load, you can do this manually or you can simply run the command: '`/usr/share/mkinitrd/mkinitrd_command_generator.sh`' with the “-l” option then pass the full path to the location of vmlinuz-generic.

```
/usr/share/mkinitrd/mkinitrd_command_generator.sh -l /boot/vmlinuz-
generic-4.4.172

image = /boot/vmlinuz-generic-4.4.172
  initrd = /boot/initrd.gz
  root = /dev/sda1
  label = 4.4.172
  read-only
# Linux bootable partition config ends
```

Now let's add this information to “/etc/lilo.conf”. Add the block of text above (starting with “image =”) at the end of lilo.conf. One way of doing that is:

```
/usr/share/mkinitrd/mkinitrd_command_generator.sh -l /boot/vmlinuz-
```

```
generic-4.4.172 >> /etc/lilo.conf
```

It is extremely important that you do NOT delete the section for the Huge kernel from `/etc/lilo.conf`, always leave it in case of tragedy with your system. . This for a possible rescue. More generally speaking: never touch the section in `/etc/lilo.conf` which defines the kernel you are running at the moment.

Using an ASCII text editor, you now change the value of the *label* statement to a better name like “Generic” or something. This text-label will be appearing in the selection menu on the Lilo screen. With a better name it is easier for you to identify and select your new kernel.
After you have done the above steps you need to run the command '`lilo`' for the new configuration to be saved (note that you always need to re-run `lilo` after you do something to your boot kernels):

```
lilo

Warning: LBA32 addressing assumed
Added Linux  *
Added Generic  +
One warning was issued.
```

Ready! Now just reboot your machine and select your “Generic” kernel.

Sources

- Original source: http://slackjeff.com.br/artigos/huge4generic.html
- Originally written by Slackjeff

howtos, slackware, kernel huge, kernel generic, author slackjeff

From:
https://docs.slackware.com/ - **SlackDocs**

Permanent link:
https://docs.slackware.com/howtos:slackware:kernel_huge_for_generic

Last update: **2021/02/22 08:47 (UTC)**