Fixes for annoyances in Slackware

USB scanner and hotplug

When you have a USB scanner attached to your computer when it boots, and the computer uses *hotplug* to initialize your hardware, you will probably see something similar to the following message:

```
chown: cannot access `/proc/bus/usb/001/005': No such file or directory
chmod: cannot access `/proc/bus/usb/001/005': No such file or directory
```

When you plug in your USB scanner *after* the computer has already booted up, the error does not appear and scanning works for non-root users. It is annoying to have to unplug the scanner before booting in order to be able to scan as a non-root user. Here's a fix.

The problem is due to an error in the `/etc/hotplug/usb.rc` script, but that can be repaired easily. Look for the following line in `/etc/hotplug/usb.rc`

```
devbus=$( ( echo -n 000 ; cat $devlink/../../devnum ) | grep -o ...\$ )
```

and change that line to

```
devbus=$( ( echo -n 000`echo $devlink| sed 's/^.*usb\([0-9]\+\)\/.*$/\1/'` ) | grep -o ...\$ )
```

After this change, either reboot the computer or restart hotplug:

```
/etc/rc.d/rc.hotplug restart
```

Now, the correct permissions and ownership are applied to the device file.

Remember to add yourself to the `scanner` group if you want to use your scanner as a non-root user! You can use the `vigr` command to add your name to the `scanner` group, but if you're uncomfortable with directly editing the system files, here is a one-liner to add user 'geek' to group 'scanner' (replace 'geek' with your own account name):

```
usermod -G $(id -Gn geek | tr ' ' ','),scanner geek
```

If the group scanner does not exist, simply create it using the command

```
groupadd scanner
```

ALSA OSS sequencer not loaded

In Slackware 10.2, the ALSA OSS-compatible sequencer module is not loaded. This results in missing device files `/dev/midi` and `/dev/sequencer`.
Edit the file `/etc/rc.d/rc.alsa` and look for the function

```
load_alsa_oss_modules() {
  if ! cat /proc/modules | grep -wq snd-pcm-oss ; then
    if ! cat /proc/modules | grep -wq snd_pcm_oss ; then
      echo "Loading OSS compatibility modules for ALSA."
      modprobe snd-pcm-oss
      modprobe snd-mixer-oss
}
```

Add the following line after the line `modprobe snd-mixer-oss` in that function:

```
      modprobe snd-seq-oss
```

and restart ALSA.

Using Samba without installing CUPS

In Slackware, CUPS is available as the default printing solution, while the old lprNG remains in the “`/pasture`” directory. Some people still prefer the trusted lpr/lpd style of printing and do not install CUPS. However, the Samba package is compiled against the CUPS libraries. Although the necessary CUPS libraries are always installed with the *aaa_elflibs* package, Samba still periodically complains loudly in the `/var/log/messages` logfile about the absent CUPS server:

```
smbd[....]: Unable to connect to CUPS server localhost - Connection refused
```

The solution (if you don't need a printing facility in Samba) is to disable printing completely. This is what you need to add to the **`[Global]`** section of your `/etc/samba/smb.conf` file:

```
load printers = no
printing = bsd
printcap name = /dev/null
```

Where is mkfs.vfat?

FAT32 (also known as vfat) partitions are commonly used on USB sticks, and they are also quite useful when you have a dual-boot system with Windows on a NTFS partition on the “other side”, and want a shared partition where both OS-es can write files.
Now, where did that “`mkfs.vfat`” command go, that seems to be available on so many other Linux distributions and is mentioned in lots of Google search results when you try to find out how to format

your partition as FAT32 instead of FAT16?

The `mkdosfs` command aka the `mkfs.msdos` command (they're the same command, one is symlinked to the other) which by default creates FAT16 (“*DOS*”) partitions, can create these FAT32 partitions as well!
Run it with the **-F32** switch:

```
mkfs.msdos -F32 /dev/<devicename>
```

The bootup messages scroll off my screen too fast

When your Linux kernel boots, it spits out all kinds of informative messages. When the Slackware init scripts start, you'll get even more messages that scroll across your screen. The kernel logs its messages in a ring buffer that you can display (after login) with the command

```
dmesg
```

If you wait too long with that command, the kernel messages that are logged after the initial boot will erase the beginning of the buffer (it has a limited capacity). You can still read the initial buffer content at your leisure though: it is saved by Slackware in the file `/var/log/dmesg`.

Still, not all of the standard output and standard error from the init scripts is logged to disk. Only when the syslogger is started, will the scripts start logging to `/var/log/messages`, `/var/log/syslog` etc… but you easily miss information if you for instance have a hardware problem. The text scrolls off the screen so quickly that you usually can't properly analyze it (unless your hardware is terribly slow 🙂).

One trick can save the day: when your computer finsishes booting up and displays the login prompt, you can scroll back by pressing the key combination Shift Page Up repeatedly!
You need to use a VESA console though, and not have your system configured for graphical login (runlevel 4).

"A security policy in place prevents this sender …" error

With Slackware 12.0 and onwards, when you are running X Window, and are greeted by the following message when you insert a CD, DVD, or USB stick into the computer:

> A security policy in place prevents this sender from sending this message to this recipient, see message bus configuration file (rejected message had interface “org.freedesktop.Hal.Device.Volume” member “Mount” error name “(unset)” destination “org.freedesktop.Hal”)

this means that you need to add your user account to the *plugdev* group.

The command to add your account (for example account called *“alien”*) to the group *plugdev* is:

```
gpasswd -a alien plugdev
```

You need to logout and login again in order for this change to have effect.
If you start your computer in runlevel 3 (non-graphical boot) and run “`groups`” you will notice that your account seems to be part of the *plugdev* group already. This is true in a sense: Slackware adds your account to this group and several others like cdrom, floppy *dynamically* for the duration of your login session. Unfortunately the DBUS/HAL daemons do not use Linux system calls to check your group membership. Instead, they rely on what is written in the `/etc/group` file.

This HAL related issue is actually explained in the CHANGES_AND_HINTS.TXT file (here is a link to the Slackware 12.1 version of CHANGES_AND_HINTS.TXT). Highly recommened reading material, that file!

From:
https://wiki.alienbase.nl/ - **Alien's Wiki**

Permanent link:
https://wiki.alienbase.nl/doku.php?id=slackware:fixes

Last update: **2008/11/18 16:00**