Devuan Management

Some are applicable to any APT-based distro.

Note: *To remove translations,*

```
$ cd /etc ; echo "Acquire::Languages { \"none\"; };" >
apt/apt.conf.d/99translation ; cd - >/dev/null
```

Note: *To automatically update system essentials,*

```
$ apt install unattended-upgrades ; dpkg-reconfigure --priority=low
unattended-upgrades
```

Upgrading

To do an upgrade:

- Modify `/etc/apt/sources.list` and replace the release codenames

  ```
  # sed -i 's/old_release/new_release/g' /etc/apt/sources.list
  ```

  - skip `-i` option for a dry-run
- Update package list

  ```
  # apt update
  ```

- Upgrade distribution

  ```
  # apt dist-upgrade
  ```

- To make sure EVERYTHING is upgraded (avoids old packages held back)
  - `# apt full-upgrade`
- Do house cleaning

  ```
  # apt autoremove
  # apt clean
  ```

That should do it!

List Installed Packages

Using `apt` tool

```
$ apt list --installed 2>/dev/null | grep installed
```

Note that apt will issue a warning when piping its output in shell. Hence, the need to redirect `stderr` to `/dev/null`. We can further grep away the packages that were automatically installed.

Using the basic `dpkg` tool

```
$ dpkg --get-selections | sed -n 's/^\([^\t]*\)\t.*$/\1/ p'
```

This version, however, only provides package name. To extract similar output from `apt` (assuming output was redirected into a file called `temp.txt`), run

```
$ cat temp.txt | sed -n 's|^\(.*\)/.*$|\1| p'
```

Adding More Repo

Get proper signature key from that source (`*.asc` file) and add to system using `apt-key`.

```
# cat <key-file.asc> | apt-key add -
```

Create a listing file for source URL in `/etc/apt/sources.list.d/`.

```
# echo "deb [arch=?] <new-repo-url> <version> main" >
/etc/apt/sources.list.d/new-repo.list
```

After doing an `apt update`, should be able to `apt install <pkg>`...

Reconfigure Package

Basically, run a `dpkg-reconfigure <pkg>`

e.g. To change timezone

```
# dpkg-reconfigure tzdata
```

Search package for specific binary

- There is a specific tool for that

  ```
  # apt install apt-file
  ```

- Update the package/file mapping database

  ```
  # apt-file update
  ```

- Search for “top”

```
# apt-file search --regexp '/top$'
```

Avoiding marking package as manually installed

- Use dry-run (simulation) option `-s`

```
# apt install -s <pg1> ... <pkgN> 2>/dev/null|grep manually
```

- Removed the package(s) displayed from your install list and re-run without `-s`

GRUB Stuffs

To prevent GRUB from looking for other OS every time it is updated:

- edit `/etc/default/grub` and insert

```
GRUB_DISABLE_OS_PROBER=true
```

- run update-grub

To add a custom entry:

- add a menuentry in `/etc/grub.d/40_custom`
- uses the same menuentry format as in `/boot/grub/grub.cfg`
  - e.g. to boot my Slackware partition I can insert

```
menuentry 'Slackware' --class slackware --class gnu-linux --class
gnu --class os $menuentry_id_option 'my1part-<uuid>' {
    insmod part_gpt
    insmod ext2
    set root='hd0,gpt4'
    if [ x$feature_platform_search_hint = xy ]; then
      search --no-floppy --fs-uuid --set=root --hint-bios=hd0,gpt4
--hint-efi=hd0,gpt4 --hint-baremetal=ahci0,gpt4  <uuid>
    else
      search --no-floppy --fs-uuid --set=root 4d95a526-2518-4fd6-
a904-f7bd2729145d
    fi
    linux /boot/vmlinuz-huge-4.4.240 root=/dev/sda4
}
```

  - use `initrd` (after `linux` line) to specify an initrd/initramfs
  - of course, <uuid> should be a valid filesystem uuid
- run update-grub

KVM Stuffs

If KVM group is missing, simply create one...

```
# [ -z "$(cat /etc/group|grep kvm)" ] && addgroup --gid 125 kvm
```

From:
https://azman.my1matrix.cc/ - **Azman @MY1**

Permanent link:
https://azman.my1matrix.cc/doku.php?id=linux:dev1_3manage

Last update: **2026/04/05 02:44**