2nd Assignment: *Application Installation*

Arnau Sangrà i Rocamora

April 17, 2015

Quines maneres d'instal.lar aplicacions coneixeu? Buscar informació sobre els següents formats de distribució d'aplicacions: tar, gz, rpm, deb i zip.
Indiqueu les comandes [i les seves opcions] que es poden fer servir per:

- Veure el contingut d'aquests tipus de fitxers

- Extreure el seu contingut, amb possibilitats d'extreure'l només parcialment

When installing applications on a UNIX system, there are several ways to accomplish the installation, each one has its advantages and drawbacks. Depending on our goals, it might be better one way than another.

- **From source code**
  Despite being more complex than the other methods, installing from source code, the user can install the application on every specific and rare system and tune the application to fit the specific user requirements. The most noticeable drawback is the installation of the application dependencies which can lead to a recursive problem or when updating the application.

- **Self-installable binaries**
  These way is mostly used by privative software. The installation suits the application, all the dependencies are bundled. Whilst ensuring the correct execution of the application it usually leads to duplication of dependencies and use of outdated libraries with discovered vulnerabilities.

- **Pre-compiled binaries**
  Applications installed from pre-compiled binaries are specially tailored for the system in general terms. Usually these are acquired from repositories for specific OS's. Despite being packages for an concrete operating system and architecture, their principal drawback is precisely that its target is the main public and the final user have not control of the compilation options used which may be particularly useful for their own system.

When downloading applications, the extension of the file/s downloaded are used to denote the format of the file. Some of the most common extensions are:

- `tar` stands for *Tape Archive* and is the most common way to distribute source code. With the unix command `tar` it is possible to manage archived files which can also be compressed in order to save space. The most important options are *-t* to list the content of a tar file, `-x` to unarchive or `-c` to archive the files and `-f` to specify the file. As mentioned before, it is possible to compress or uncompress the tar files by using different options such as to *gzip* (`-z`) or *bzip2* (`-j`) programs. One way to specify the files to be extracted is by adding `--wildcards` and `--no-anchored` arguments followed by an escaped regex with the files to extract pattern.

- `rpm` refers to RPM Package Manager, initially created for Red Hat OS, and later incorporated in other Linux distributions such as SuSE or Mandriva and files with .rpm extension.

        `-i` package_name to install the package.

        `-l` lists files of the package

        `-c` list only configuration files (implies -l).

        `-d` list only documentation files (implies -l).

- `deb` is a file extension used by Debian binary packages. These files are usually handled by the program `dpkg` from Debian and other OS based on that. Files with this extension are no different than other Unix archives. Internally they store two tar files one for the control information and the other one actually contains de program data.

- `pacman` is the main package manager of *Arch Linux*. Its goal is to simplify the installation of packages and its dependencies. Pacman can install either from the official Arch Linux Repository or packages build by the own user, i.e. from the AUR (Arch User Repository). Written in C, it handles packages withwith pkg.tar.xz extension. Depending on the first character of the option, pacman will query the local installed packages `-Q` or the package repository `-S`. Some of the most important features are achieved with the following options:

        `-Qs package_name` Search the installed packages database for certain package.

        `-Ss package_name` Search for a pattern in the package repository in both name and description fields.

        `-S package_name` Install the package from the repositories that matches the provided argument.

        `-Qen` List all the user installed packages.

        `-Syu` Synchronize package database and update the system to latest version.