

TP 3A - GNU/Linux - Installation de Logiciels

Nicolas Delanoue

Exercice 1 (Compression et décompression)

1. Faites une sauvegarde de tous les fichiers d'un répertoire de test sous forme d'une archive compressée (que l'on pourra nommer `archive.tar.gz`).
2. Avec la commande `tar`, faites une sauvegarde de ce même répertoire de test sous forme d'une archive *non compressée*.
3. Comparez la taille de ces deux fichiers d'archives avec la commande `du -h`. A quoi correspondent les options `h` et `s` de la commande `du`? Testez.

Exercice 2 (Installation de GNOME Tetravex avec l'application Logiciel)

1. Lancez l'application Logiciel (depuis le menu).
2. Dans les jeux, trouvez Tetravex et installez-le.

Exercice 3 (Installation de gparted avec l'application Logiciel d'Ubuntu)

1. Lancez l'application Logiciel (dans le menu).
2. Dans la zone de recherche, recherchez `gparted`. Dans la liste au-dessous, double-cliquez sur l'entrée qui apparaît.
3. A quoi sert ce logiciel?
4. Installez `gparted`,
5. Lancez-le, combien de partition compte votre disque dur?
6. désinstallez-le puis fermez la logithèque.

Exercice 4

1. Utilisez `apt` pour installer `gparted`. Au cours du processus d'installation, soyez attentif à ce qui s'affiche! Existe-t-il des dépendances pour `gparted`?
2. Après l'installation, utilisez la commande permettant d'afficher la liste des dépendances de `gparted`: cela devrait confirmer ce que vous avez noté à la question précédente.
3. Désinstallez `gparted`.

Exercice 5 (Installation des outils de développement)

1. En une seule commande `apt`, installez `gcc`, `make` et `autoconf`,
2. Utilisez la commande `apt-cache show` pour obtenir la description du paquet `gcc` puis du paquet `make`.

Exercice 6

1. Utilisez `apt-cache` pour chercher tous les paquets dont le nom contient `codeblocks`.
2. Installez `codeblocks` avec `apt`. Au cours du processus d'installation, soyez attentif à ce qui s'affiche! Existe-t-il des dépendances pour `codeblocks`?

3. Lancez `codeblocks`.
4. Utilisez cet éditeur pour visualiser un fichier contenant du langage `c`. On pourra par exemple regarder :
http://perso-laris.univ-angers.fr/~delanoue/polytech/gnu_linux_python/main.c
5. Fermez `codeblocks`.
6. Utilisez `apt` pour désinstaller le paquet `libcodeblocks0` (faites attention à l'orthographe, sinon ça ne marchera pas !) et soyez attentif à ce qui s'affiche : existe-t-il des suppressions pour dépendance ? Pourquoi ?
7. Est-ce que les autres dépendances de `codeblocks` ont été supprimées, elle ? Pourquoi ?
8. Désinstallez-les avec `apt autoremove`.

Exercice 7 (Installation de gnuchess via un fichier deb)

Dans cet exercice, on utilise la commande `dpkg`.

1. Trouvez, dans les dépôts d'ubuntu avec votre navigateur web préféré, le paquet `deb gnuchess`. Attention, en fonction de votre distribution et de votre architecture matérielle, choisissez le bon fichier `deb` ...
2. Téléchargez-le.
3. Avant de l'installer : affichez ses informations.
4. Installez-le.
5. Désinstallez le paquet.

Exercice 8 (Installation de ClassicLadder)

1. `ClassicLadder` est un logiciel de simulation de grafquets. Téléchargez-en les sources et décompressez-le dans votre répertoire de base.
2. Visualisez son `Makefile`.
3. Compilez les sources avec votre compte utilisateur.
4. Vérifiez qu'il est fonctionnel.

Exercice 9 (Première compilation)

1. Dans un répertoire nommé `premier_programme`, créez un fichier `main.c` dont le contenu est le suivant :

```

1 #include <stdio.h>
2 int main() {
3     printf ("Hello, World!\n");
4     return 0;
5 }
```

2. Compilez ce programme dans votre répertoire avec la commande suivante :
`gcc main.c`
3. Quel fichier est apparu dans votre répertoire ?
4. Quels sont les permissions de ce dernier ?
5. Exécutez ce binaire.
6. Modifiez la commande de la question 2 pour le binaire précédente se nomme `hello.exe`.

Exercice 10 (Installation de 7zip)

1. 7zip existe aussi pour Linux sous le nom du projet p7zip. Rendez-vous sur le site de p7zip (sur sourceforge.net) et téléchargez les sources de p7zip.
2. Décompressez l'archive dans `/usr/local/src` : y a-t-il un `Makefile` ? Est-il nécessaire de faire un `./configure` ?
3. Compilez les sources puis installez les binaires (utilisez les commandes vues en cours).
4. Faites un copier-coller de la liste des fichiers installés.
5. Dans quel répertoire les binaires ont-ils été installés ?
6. Faites le ménage : supprimez manuellement tout ce qui a été installé (utilisez pour cela la liste obtenue deux questions plus tôt).