



Université Abdelmalek Essaadi

École Nationale des Sciences Appliquées Al-Hoceima

Projet :

**Développement d'un système de fichiers et d'une
distribution Linux personnalisée**

Première Année Ingénierie des données

Réalisé par :

- **EL GUELTA MOHAMED-SABER**
- **ELHADIFI SOUKAINA**
- **IBNCHAKROUNE HOUSSAM**
- **KAMAL SALMA**

Encadré par : Pr. Mohamed CHERRADI

Année Universitaire : 2023/2024

Remerciement

Nous tenons à exprimer notre sincère gratitude envers notre encadrent Mr. Mohamed Cherradi pour avoir bien voulu proposer le sujet de notre travail de recherche et pour l'encadrement attentif qu'il nous a prodigué tout au long de ce projet. Ses conseils précieux ont été d'une aide inestimable, et nous sommes reconnaissants de la chance que nous avons eu de bénéficier de son expertise.

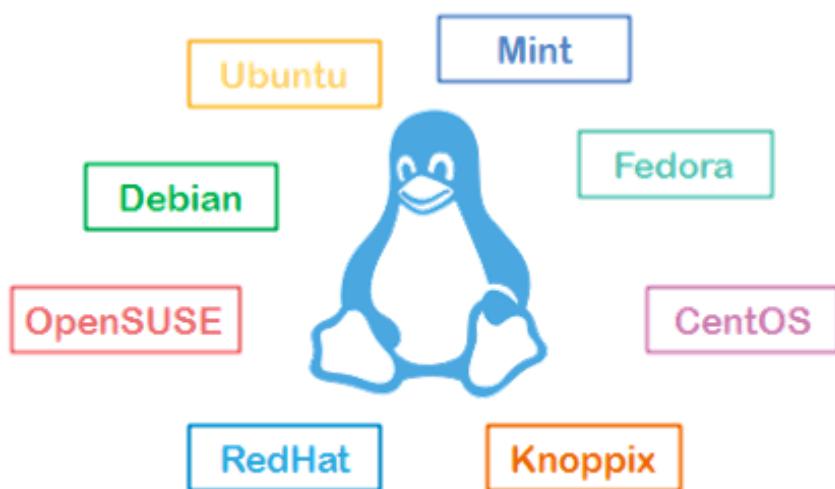
Ce travail a représenté pour nous bien plus qu'une simple réalisation académique. Il nous a permis d'acquérir des compétences essentielles, tant sur le plan technique que sur le plan logique. Nous avons appris à résoudre des problèmes complexes et à développer notre maîtrise du système d'exploitation Linux. Ces apprentissages sont des acquis qui seront indéniablement utiles dans notre parcours académique et professionnel futur.

Introduction

Linux est un système d'exploitation de type Unix qui a été créé en 1991 par Linus Torvald, un étudiant finlandais en informatique. Il est aujourd'hui l'un des systèmes d'exploitation les plus utilisés au monde, surtout dans le domaine des serveurs et des systèmes embarqués.

Distribution Linux :

Une distribution Linux, est une version spécifique qui regroupe le noyau Linux, les utilitaires système, les bibliothèques, un ensemble d'applications logicielles et des gestionnaires de paquets



Arborescence PTFS :

Lors du développement d'une distribution Linux personnalisée, nous nous sommes basés sur la structure hiérarchique du système de fichiers de Linux en créant une arborescence sous le nom '**PLANTFS (PTFS)**', où le point de départ (la racine) est appelé ptfs, et tous les autres répertoires et fichiers se trouvent en dessous de ce répertoire. Cette structure nous a fourni une base organisée pour le placement des fichiers système, des programmes utilitaires, des fichiers de configuration, et d'autres éléments essentiels à une distribution Linux.

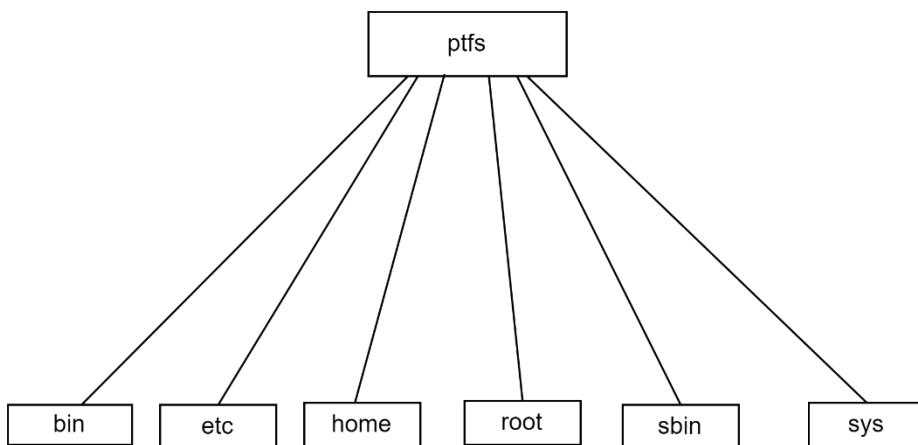


Figure 4 : arborescence PTFS

bin : ce répertoire contient les scripts des commandes et les fichiers exécutables.

etc : un répertoire qui contient des informations sur les utilisateurs, y compris les mots de passe.

home : c'est l'emplacement par défaut des répertoires personnels des utilisateurs.

root : le répertoire personnel du super utilisateur(root).

sbin : contient les scripts des commandes et les fichiers exécutables avec des priviléges d'administration.

sys : ce répertoire contient les informations sur l'état de système.

Les points forts de PTFS :

Les points forts du système PTFS résident dans son approche fondamentale qui tire parti de la structure de l'arborescence Linux :

- Perspective visuelle sur la structure du système de fichiers, facilitant la compréhension de la disposition des fichiers et répertoires.
- Expérience de navigation robuste et flexible, permettant aux utilisateurs de se déplacer avec aisance à travers la hiérarchie des répertoires, et d'accéder directement aux répertoires spécifiques en utilisant des chemins absous ou relatifs, créant ainsi une expérience utilisateur cohérente avec les normes Linux établies.
- Notion du supers utilisateurs (**sudoers**), permettant aux utilisateurs d'exécuter des commandes avec des privilèges administratifs, assurant ainsi plus de sécurité.

TABLE DES MATIÈRES

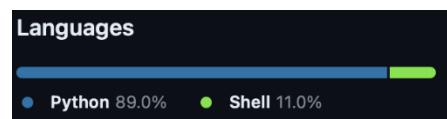
I- Outils de développement.....	7
1. Langages de programmation.....	
2. Plateformes de développements.....	
3. Bibliothèques.....	
II- Commandes PTFS.....	9
III- Interface graphique.....	11
IV- Génération d'une image iso avec l'outil Cubic.....	13

1- Outils de développement :

Pour développer une distribution Linux personnalisée nous avons utilisé divers outils :

1-1 Les langages de programmation :

- Shell Bash
- Python
- Pourcentage selon github



1-2 Les plateformes de développement :

- Ubuntu
- VS Code



Visual Studio Code

1-3 Bibliothèques :

os : Gère les interactions avec le système d'exploitation, l'accès au système de fichiers et la manipulation des variables d'environnement

json : Manipule, lit et écrit des données au format JSON (JavaScript Object Notation), un format de données largement utilisé.

argparse : Facilite l'analyse des arguments de ligne de commande pour créer des interfaces conviviales pour les scripts Python.

hashlib : Fournit des algorithmes de hachage sécurisés pour le cryptage des données sensibles.

getpass : Permet de demander de manière sécurisée un mot de passe à l'utilisateur sans afficher le texte saisi à l'écran.

time : Gère les fonctionnalités liées à la mesure du temps, à la gestion du temps et à la manipulation de la date et de l'heure

Sys : fournit un accès à certaines variables utilisées ou maintenues par l'interpréteur Python, ainsi que des fonctionnalités liées au système.

Pygame : pygame est une bibliothèque de jeux open source pour Python. Elle permet de créer des jeux et des applications multimédias

Bz2 permet la compression et la décompression de fichiers au format bzip2

Gzip : permet la compression et la décompression de fichiers au format gzip.



2- Commandes de PTFS :

Puisque nous avons été baser sur les distributions de Linux, nous avons essayé d'intégrer des commandes familières telles que 'ptfs_tree' et 'ptfs_ls' pour une visualisation claire de la structure du système de fichiers, 'ptfs_sudo' pour les permissions et les tâches administratives, et d'autres plusieurs commandes.

Le tableau suivant englobe toutes les commandes de **ptfs** et leurs descriptions :

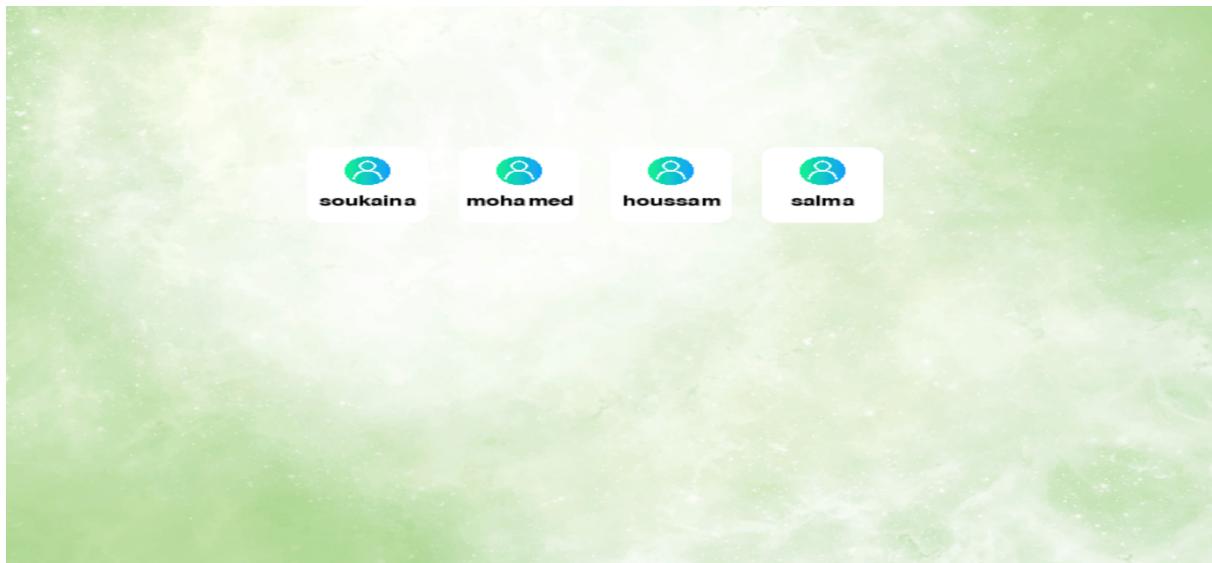
Remarque : L'option [-h] ou [--help] permet d'obtenir des informations sur l'utilisation d'une commande particulière et pour explorer les options disponibles avec chaque commande.

Commande[option]	Description
ptfs [on] [off]	Démarrer ou arrêter le système
ptfs_tree	Afficher la structure d'arborescence
ptfs_cd	Changer le répertoire de travail courant
ptfs_pwd	Afficher le répertoire de travail actuel
ptfs_who [-i] [-o]	Afficher la liste des utilisateurs dans le système [-i] : Afficher l'utilisateur connecter [-o] : Afficher les utilisateurs non connecter
ptfs_su	Se connecter à un autre utilisateur
ptfs_addusr	Ajouter un utilisateur
ptfs_delusr	Supprimer un utilisateur
ptfs_usrmmod [-u] [-p]	Modifier les informations d'un utilisateur [-u] : Modifier le nom [-p] : Modifier le mot de passe
ptfs_whos	Afficher la liste des superutilisateur
ptfs_sudo	Exécuter des commandes avec des privilèges d'administration
ptfs_addsudo	Ajouter un superutilisateur
ptfs_delsudo	Supprimer un superutilisateur
ptfs_mkdir [--comment]	Créer un dossier vide [--comment] : ajouter un commentaire
ptfs_rmdir [-r]	Supprimer un dossier vide ; [-r] : supprimer le dossier et ses fichiers

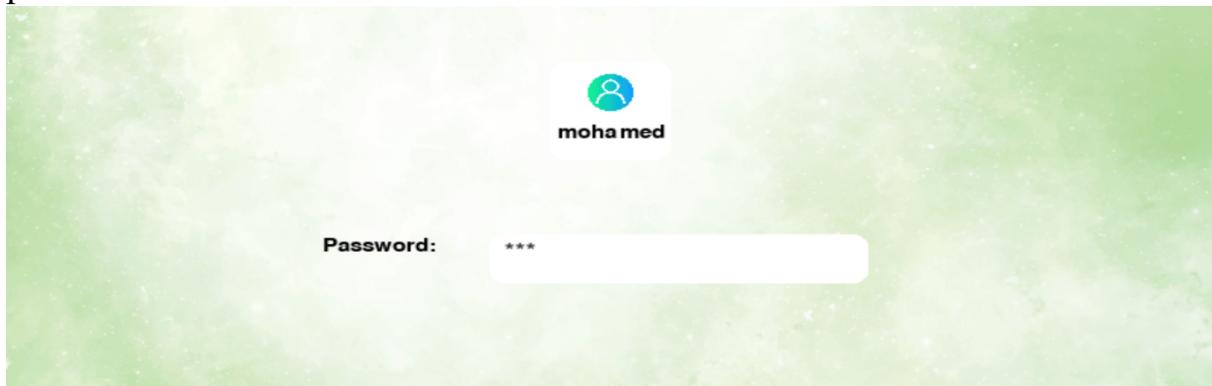
ptfs_touch [--comment]	Créer un fichier [--comment] : ajouter un commentaire
ptfs_rm	Supprimer un fichier
ptfs_type	Afficher le type, soit fichiers ou dossier
ptfs_ls [-l] [-H]	Lister les fichiers et les dossiers d'un répertoire [-l] : ajouter des détails [-H] : lister les fichiers cachés
ptfs_cat [-l] [-H] [-t]	Afficher le contenu d'un fichier [-l] : afficher les 10 premières lignes [-H] : les 3 premières lignes [-t] : les 3 dernières lignes
ptfs_edit [-v] [-n]	Éditer les fichiers texte [-v] avec l'éditeur de texte vim [-n] avec l'éditeur de texte nano
ptfs_metadata	Afficher les métas data d'un fichier ou d'un dossier
ptfs_zip [-g] [-b]	Compresser les fichiers et les dossiers [-g] sous format gzip [-b] sous format bzip2
ptfs_ip [-pr] [-pu]	Afficher l'adresse IP du système [-pr] : adresse privée [-pu] : adresse publique
ptfs_ps	Afficher des informations sur les processus actuels.

3- Interface graphique de ptfs

Pour offrir une expérience interactive aux utilisateurs, nous avons intégré une interface graphique à notre système de fichiers PTFS. Lors de l'exécution de la commande 'ptfs_gui', l'utilisateur est invité à choisir son compte parmi une liste d'options disponibles.



Une fois que l'utilisateur a sélectionné un compte, le système sollicite la saisie du mot de passe associé à ce compte. Une étape cruciale de la procédure d'authentification, si le mot de passe fourni n'est pas correct, l'utilisateur ne sera pas autorisé à accéder à son bureau.



En cas de saisie correcte du mot de passe, l'utilisateur est redirigé vers son bureau personnel. Ce bureau offre une interface conviviale qui comprend un gestionnaire de fichiers permettant la manipulation des fichiers et des dossiers associés au compte. De plus, un terminal spécifique à l'utilisateur est disponible, offrant un accès direct aux fonctionnalités du système d'exploitation.



Le bureau de chaque utilisateur est équipé d'un explorateur de fichiers permettant une gestion intuitive des ressources. Le terminal associé facilite l'exécution de commandes spécifiques au compte. En outre, une fonction d'arrêt est mise à disposition pour permettre à l'utilisateur de quitter son environnement.

4- Génération d'une image ISO avec cubic

En utilisant une application nommée Cubic (**Custom Ubuntu ISO Creator**), c'est un outil qui permet de personnaliser et de créer une images ISO personnalisées d'Ubuntu.

Les étapes d'utilisation de Cubic :

Pour l'installation il faut exécuter les commandes suivantes une par une dans le terminal :

1. ajouter un dépôt PPA (Personal Package Archive) au système Ubuntu

```
| sudo apt-add-repository ppa:cubic-wizard/release
```

2. importer la clé GPG associée à ce dépôt.

```
| sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys 6494C6D6997C215E
```

3. Mettre à jour le système et installer Cubic.

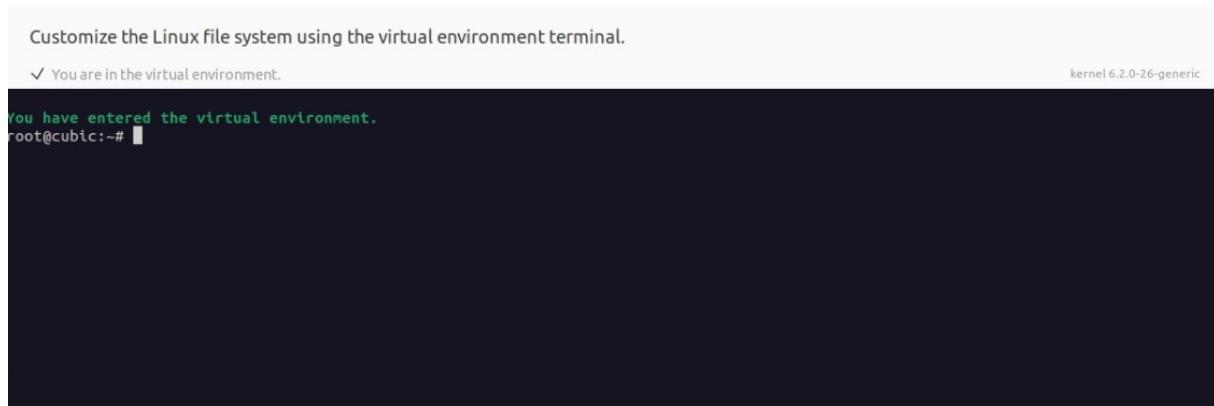
```
| sudo apt update && sudo apt install cubic
```

4. Démarrer l'application



Remarque : /home/user/iso_ptfs est le chemin du répertoire dont on sauvegarde les fichiers du projet.

5. Ouvrir environnement Chroot



Customize the Linux file system using the virtual environment terminal.

✓ You are in the virtual environment.

You have entered the virtual environment.
root@cubic:~#

kernel 6.2.0-26-generic

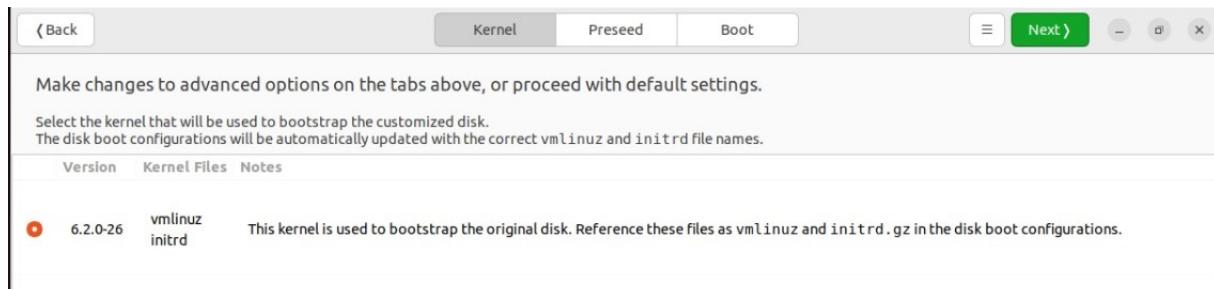
C'est la partie où on personnalise notre image ISO :

- ❖ on crée un utilisateur appelé ptfs avec un mot de passe en utilisant la commande :

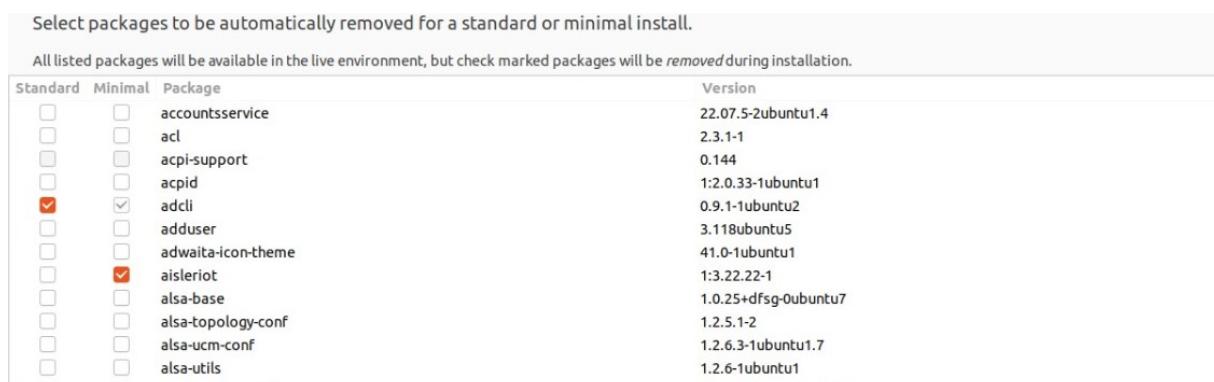
| adduser ptfs

- ❖ on ajoute notre file system ptfs dans cet utilisateur

6. Choisir la version du kernel



7. Sélectionner les paquets à supprimer automatiquement après l'installation

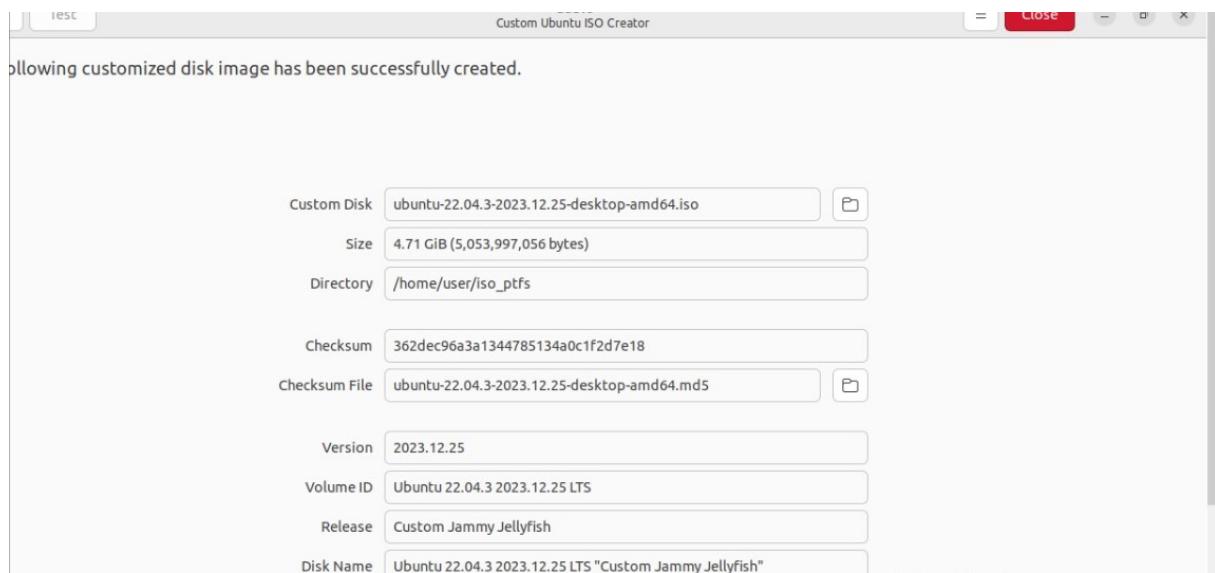


Select packages to be automatically removed for a standard or minimal install.

All listed packages will be available in the live environment, but check marked packages will be removed during installation.

Standard	Minimal	Package	Version
<input type="checkbox"/>	<input type="checkbox"/>	accounts-service	22.07.5-2ubuntu1.4
<input type="checkbox"/>	<input type="checkbox"/>	acl	2.3.1-1
<input type="checkbox"/>	<input type="checkbox"/>	acpi-support	0.144
<input type="checkbox"/>	<input type="checkbox"/>	acpid	1:2.0.33-1ubuntu1
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	adcli	0.9.1-1ubuntu2
<input type="checkbox"/>	<input type="checkbox"/>	adduser	3.118ubuntu5
<input type="checkbox"/>	<input type="checkbox"/>	adwaita-icon-theme	41.0-ubuntu1
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	aisleriot	1:3.22.22-1
<input type="checkbox"/>	<input type="checkbox"/>	alsa-base	1.0.25+dfsg-0ubuntu7
<input type="checkbox"/>	<input type="checkbox"/>	alsa-topology-conf	1.2.5.1-2
<input type="checkbox"/>	<input type="checkbox"/>	alsa-ucm-conf	1.2.6.3-1ubuntu1.7
<input type="checkbox"/>	<input type="checkbox"/>	alsa-utils	1.2.6-1ubuntu1

8. Création de l'ISO



Notre image est donc prête à utiliser.

end !