



Rapport de Projet

Première Année Ingénierie des données

ID-1
DISTRIBUTION

Réalisé par :

- **BOUKAICH Mohammed Rida**
- **LAMADI Youssef**
- **KBIRI-ALAOUI Abdelhafid**
- **SABBAR Mohamed**

Encadré par :

- **Pr. CHERRADI MOHAMED**

SOMMAIRE :

I. Introduction

1. Définition générale de distribution linux
2. Définition générale de système de fichiers
3. La relation entre système de fichiers et la distribution

II. Outils utilisés

1. Python
2. La distribution de base
3. Cubic

III. File system

1. Architecture de file system.
2. Gestion des fichiers
3. Corbeille
4. Gestion des utilisateurs
5. Logging
6. Métadonnées

IV. Intégration de file system avec Ubuntu.

V. Guide d'utilisation.

Introduction

1. Définition générale de distribution linux

Une distribution Linux, souvent appelée "**distro**", est une version spécifique du système d'exploitation Linux, qui comprend le noyau Linux, des utilitaires systèmes, des bibliothèques et souvent un gestionnaire de paquets. Les distributions Linux sont souvent personnalisées par différentes équipes pour répondre à des besoins spécifiques, ce qui donne lieu à une variété de distributions adaptées à divers contextes, tels que les serveurs, les postes de travail, les systèmes embarqués ...

2. Définition générale de file system

Le système de fichiers est une structure organisationnelle utilisée par les systèmes d'exploitation pour stocker, organiser et accéder aux fichiers sur un disque ou un autre support de stockage. Il définit la manière dont les données sont stockées, nommées, organisées en répertoires, et accédées par le système d'exploitation et les utilisateurs.

3. La relation entre file system et distribution

La distribution Linux inclut généralement un ensemble prédéfini de logiciels, y compris un noyau Linux, des utilitaires systèmes, des bibliothèques, des outils en ligne de commande, etc. Le système de fichiers est un composant essentiel de la distribution, car il spécifie la manière dont les fichiers et les répertoires sont organisés sur le disque. Les distributions Linux utilisent souvent des systèmes de fichiers standards tels qu'Ext 4, XFS, Btrfs, etc.

Outils utilisés

1. Python

On a utilisé langage de programmation python, un langage de programmation populaire et polyvalent qui peut être utilisé dans de nombreux domaines, y compris le développement de logiciels, la data science et l'automatisation de tâches. Il est connu pour sa syntaxe claire et concise, ce qui le rend facile à apprendre et à utiliser.

Aussi ces nombreuses bibliothèques, qui nous permettent de coder la plupart des scripts de notre file system.

Pour Python on utilise plusieurs bibliothèques :



Os : offre des fonctions pour interagir avec le système d'exploitation. Elle permet de manipuler les chemins de fichiers, de travailler avec des répertoires, d'obtenir des informations système, etc.

Sys : offre des fonctionnalités utiles pour interagir avec l'environnement d'exécution Python et pour effectuer certaines opérations de bas niveau. Elle est souvent utilisée pour manipuler des paramètres système, des chemins, des exceptions et d'autres aspects liés à l'exécution des scripts Python

Shutil : fournit des fonctionnalités pour copier, déplacer, renommer et supprimer des fichiers et des répertoires, ainsi que pour gérer des opérations d'archivage et d'exploration des arbres de fichiers

Argparse : permet de simplifier le processus de création des interfaces en ligne de commande pour les scripts Python. Elle facilite la gestion des arguments passés à un script via la ligne de commande, ce qui permet de créer des scripts plus flexibles et interactifs.

Datetime : fournit des classes pour manipuler les dates, les heures et les durées. Il permet de travailler avec des objets de date et d'heure, de formater les dates, de réaliser des opérations arithmétiques sur les dates et de manipuler des intervalles de temps.

JSON : est utilisé pour travailler avec des données au format JSON (JavaScript Object Notation). JSON est un format de données largement utilisé pour échanger des données entre serveurs et clients, ainsi que pour stocker des données structurées de manière simple et lisible par l'homme.

2. La distribution de base

Linux (distribution Ubuntu) est un système d'exploitation open source qui est souvent utilisé pour les serveurs et les ordinateurs de bureau il est connu pour sa stabilité, sa sécurité et sa flexibilité et est largement utilisé dans les entreprises et les institutions académiques. On a l'utilise à cause de faciliter en fonctionnement, aussi à cause d'avoir un outil qui me permet de faire une distribution personnaliser qui on va l'explique en l'axe suivant qui est Cubic.



3. Cubic

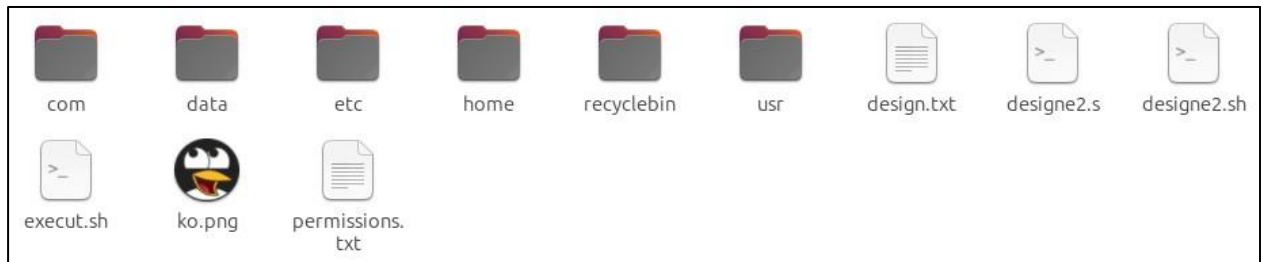
Cubic est en effet un outil spécifique à Ubuntu q qui permet de personnaliser les images ISO d'installation d'Ubuntu. Avec Cubic, vous pouvez créer une image ISO personnalisée d'Ubuntu, ce qui peut être utile dans divers scénarios, notamment pour déployer des installations personnalisées, préconfigurées ou pour créer des images d'installation avec des modifications spécifiques.



File system

1. Architecture de file system

Le file system de la distribution 'id1distro' est basé sur l'arborescence de la plupart des systèmes d'exploitation Linux sur le concept de tout est fichier et le file system est organisé dans un fichier appelé 'ID1FS'.



Ce système des fichiers est constitué sur :

Com : le répertoire qui contient les scripts des commandes.

Data : répertoire qui contient des fichiers txt et json qui contiennent des métadonnées pour le super-utilisateur (root), et écrit le logging.

Home : le répertoire contient les utilisateurs créés en ce système.

Recyclebin (exécuté en Ubuntu desktop) : répertoire qui contient les fichiers et dossiers que l'utilisateur veut supprimer et se trouvent temporairement avant de les supprimer.

Etc : répertoire qui fait le même fonctionnement de data mais il écrit des métadonnées pour le simple utilisateur.

2. Gestion des fichiers

Pour cette partie on a fait plusieurs commandes qui font cette gestion :

Crtf (touch en Ubuntu) : permet de créer un fichier.

Crtcd (mkdir en Ubuntu) : permet de créer un répertoire

Openf (cat en Ubuntu) : ouvrir un fichier.

List (ls en Ubuntu) : lister le contenu d'un répertoire

Delbin (rm en Ubuntu) : Permet de déplacer les fichiers vers la corbeille.

L'option -f : Supprimer un fichier ou un répertoire de façon permanente.

L'option -d : déplacer les répertoires vers la corbeille

NB

Chaque utilisateur a le droit de supprimer sauf ses propres fichiers, à l'exception du compte 'root'.
Chaque utilisateur a le droit de voir/supprimer sauf ses propres fichiers.

```
"Name": "file1",  
"Type": "file",  
"Owner": "root",  
"Size": 36,  
"Location": "/ID1FS/com/../../home/root/file1",  
"Date Created": "2023-12-24 16:03:15",  
"Permissions": "-rw-rw-r--",  
"Status": "Available"
```

```
root@id1distribution:/# openf file1  
Error: file 'file1' not found or permission denied
```

En connectant en 'le root' :

```
root@id1distribution:/# signin  
Enter username: root  
Enter password:  
Connected successfully as 'root'.
```

```
connected successfully as 'root'.  
root@id1distribution:/# openf file1  
Enter the text editor command (e.g., nano, vim, gedit): nano
```

3. Corbeille

NB : (une idée en la période de test exécuté bien mais en la nouveau distribution on a trouvé des problèmes).

Dltbin est une commande conçue pour gérer les opérations de fichiers en permettant aux utilisateurs de déplacer des fichiers ou répertoires sélectionnés vers la corbeille. Elle facilite à la fois les processus de suppression et de restauration.

Cette commande permet aux utilisateurs de restaurer des fichiers ou répertoires sélectionnés à leurs emplacements d'origine. De plus, elle gère efficacement l'espace disque en supprimant définitivement les fichiers dont la date de suppression dépasse 30 jours.

Pour cette corbeille on a deux répertoires :



Recyclebin ---> Endroit pour stocker les fichiers supprimés.

recyclebin/.cache ---> Contient les informations sur les fichiers supprimés (date de suppression et chemin d'accès).

Les options valables pour cette corbeille :

Dltbin	Exécutable principal et supprime les fichiers donnés
dltbin -d/--del	Supprime les fichiers ou répertoires
dltbin -r/--restore	Restaure les fichiers/répertoires donnés
dltbin -e/--emptybin	Vide la corbeille
dltbin -s/--SetAutoclean	Options : on, off, Période_en_jours pour nettoyer la corbeille
dltbin -h/--help	Affiche les informations sur dltbin
dltbin -f/--force	Supprime définitivement le fichier/répertoire donné
dltbin -l/--list	Liste le contenu de la corbeille avec l'âge depuis le jour de la suppression
dltbin -v/--version	Affiche la version actuelle du package

4. Gestion des utilisateurs

Pour cette partie on a fait plusieurs commandes qui font cette gestion :

- ✓ **signup** : créer un compte utilisateur, en le fichier json les informations de nouveau utilisateur est écrit comme cette forme.

```
{
  "username": "root",
  "hashed_password": "9af15b336e6a9619928537df30b2e6a2376569fcf9d7e773eccede65606529a0",
  "created_at": "2023-12-21 19:05:09"
},
{
  "username": "ahmed",
  "hashed_password": "9af15b336e6a9619928537df30b2e6a2376569fcf9d7e773eccede65606529a0",
  "created_at": "2023-12-21 19:06:25"
}
```

- ✓ **signin** : connexion au compte utilisateur.
- ✓ **dltusr** : supprimer un compte utilisateur, l'utilisateur supprime aussi de métras donnés.

```

    "created_at": "2023-12-24 13:02:28"
  }
]root@id1distribution:/ID1FS/data# signup mohamed
Enter password:
User 'mohamed' created successfully.
Home folder created at: /ID1FS/home/mohamed
root@id1distribution:/ID1FS/data# cat userdata.json
[
  {
    "username": "root",
    "hashed_password": "9af15b336e6a9619928537df30b2e6a2376569fcf9d7e773eccede65606529a0",
    "created_at": "2023-12-24 13:02:28"
  },
  {
    "username": "mohamed",
    "hashed_password": "5994471abb01112afcc18159f6cc74b4f511b99806da59b3caf5a9c173cacfc5",
    "created_at": "2023-12-25 02:20:47"
  }
]

root@id1distribution:/ID1FS/data# dltusr mohamed
User 'mohamed' deleted successfully.
Home directory '/ID1FS/home/mohamed' deleted successfully.
root@id1distribution:/ID1FS/data# cat userdata.json
[
  {
    "username": "root",
    "hashed_password": "9af15b336e6a9619928537df30b2e6a2376569fcf9d7e773eccede65606529a0",
    "created_at": "2023-12-24 13:02:28"
  }
]
root@id1distribution:/ID1FS/data# signin mohamed
Enter username: mohamed
Enter password:
User 'mohamed' not found.
root@id1distribution:/ID1FS/data# _

```

5. Logging

Le logging est un processus qui permet d'enregistrer les informations sur notre file system, telles que la création, la suppression et modification des fichiers et des comptes des utilisateurs. En conséquence, le logging est un outil essentiel pour comprendre et gérer l'activité d'un système des fichiers. Il peut aider à résoudre les problèmes améliorer les performances.

```

The user mohamed is created at 2023-12-25 10:07:19
The user mohamed failure to connect at 2023-12-25 10:08:34
The user mohamed connect at 2023-12-25 10:08:49
test.txt is created at 2023-12-25 10:11:19 with size 0 bytes
test.txt file is deleted at 2023-12-25 10:11:39, created on 2023-12-25 10:11:19
root@id1distribution:/ID1FS/data#

```

6. Métadonnées

Les métadonnées sont des informations qui décrivent ou qui s'appliquent à d'autres données. Elles permettent de fournir un contexte aux données et de les rendre plus faciles à comprendre et à utiliser. Par exemple, les métadonnées d'un fichier peuvent inclure son propriétaire, sa date de création, sa taille et sa location.

```

The user mohamed is created at 2023-12-25 10:07:19
The user mohamed failure to connect at 2023-12-25 10:08:34
The user mohamed connect at 2023-12-25 10:08:49
test.txt is created at 2023-12-25 10:11:19 with size 0 bytes
test.txt file is deleted at 2023-12-25 10:11:39, created on 2023-12-25 10:11:19
root@id1distribution:/ID1FS/data#

```

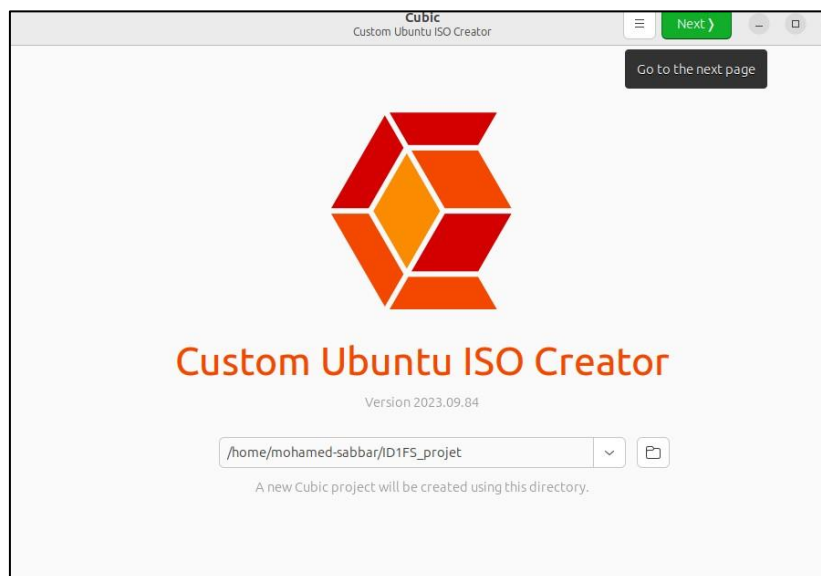

Intégration de file system avec Ubuntu

Comme on a dit précédemment Cubic permet de modifier une image ISO d'une distribution Linux basée sur Ubuntu en offrant une interface graphique conviviale pour effectuer des personnalisations. Voici une explication du processus de modification d'un fichier ISO avec Cubic :

1. Télécharger la version d'Ubuntu sur laquelle vous souhaitez baser votre distribution (Desktop ou server) pour notre projet on a travaillé avec la version server.
2. Télécharger et installer cubic (terminal d'Ubuntu) : `sudo apt install cubic`.

```
mohamed-sabbar@mohamed-sabbar-1-2:~$ sudo apt-get update
[sudo] Mot de passe de mohamed-sabbar :
Atteint :1 http://ma.archive.ubuntu.com/ubuntu mantic InRelease
Atteint :2 http://security.ubuntu.com/ubuntu mantic-security InRelease
Atteint :3 http://ma.archive.ubuntu.com/ubuntu mantic-updates InRelease
Atteint :4 http://ma.archive.ubuntu.com/ubuntu mantic-backports InRelease
Atteint :5 https://ppa.launchpadcontent.net/cubic-wizard/release/ubuntu mantic
nRelease
Lecture des listes de paquets... Fait
mohamed-sabbar@mohamed-sabbar-1-2:~$ sudo apt-get install cubic
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
cubic est déjà la version la plus récente (2023.09-84-release-202309300046-ubu
u23.10.1).
0 mis à jour, 0 nouvellement installés, 0 à enlever et 37 non mis à jour.
```

3. Lancement d'interphase du cubic : `sudo cubic`.



4. Création ou ouverture d'un projet
5. Sélection de l'image ISO de base (base sur Ubuntu server).

<

home mohamed-sabbar

📁 Bureau

>

Nom	Taille	Type	Modifié
📁 ubuntu-22.04.3-live-server-amd64.iso	2,1 Go	image CD brute	sam.
📁 ubuntu-23.10.1-desktop-amd64.iso	5,2 Go	image CD brute	3 nov.

5.1 Copier les répertoires du système des fichiers en custom-root de notre distribution.

```
mohamed-sabbar@mohamed-sabbar-1-2:~$ sudo cp -r ID1FS /home/mohamed-sabbar/ID1FS_projet/custom-root
mohamed-sabbar@mohamed-sabbar-1-2:~$
```

5.2 Personnalisation de l'image : pour cette étape on a concentré sur la transformation du répertoire com de notre système des fichiers à un variable d'environnement en notre distribution.

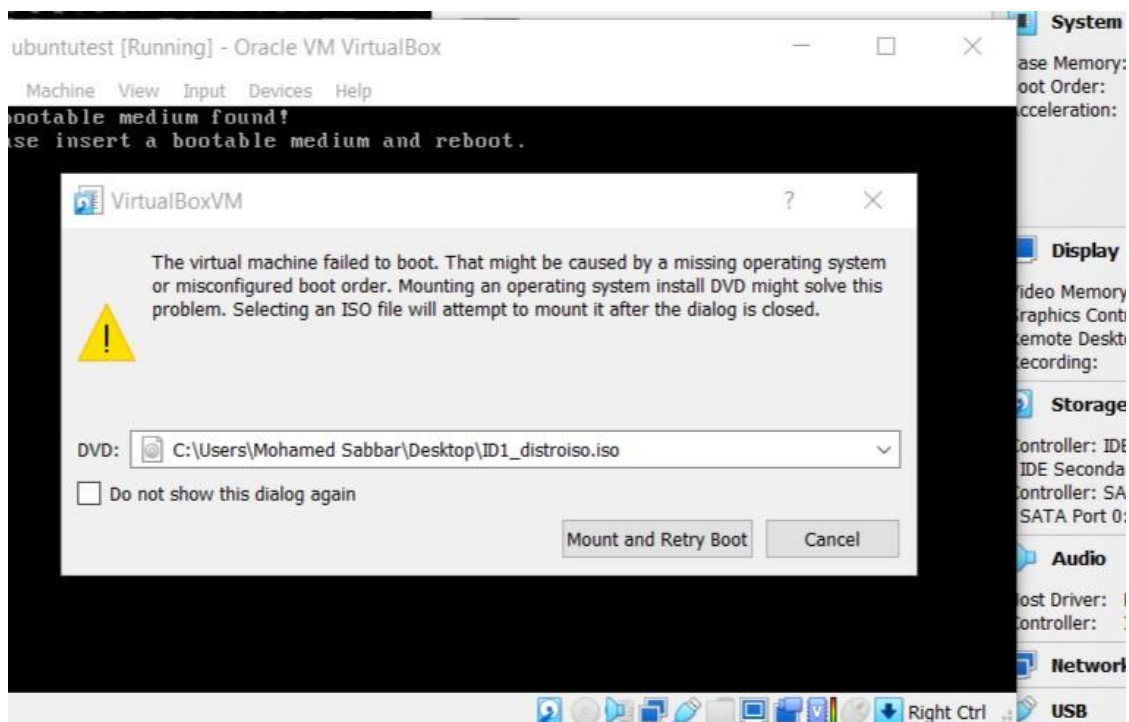
```
You have entered the virtual environment.
root@cubic:~# echo 'export PATH=/ID1FS/com' >> ~/.bashrc
root@cubic:~# source ~/.bashrc
```

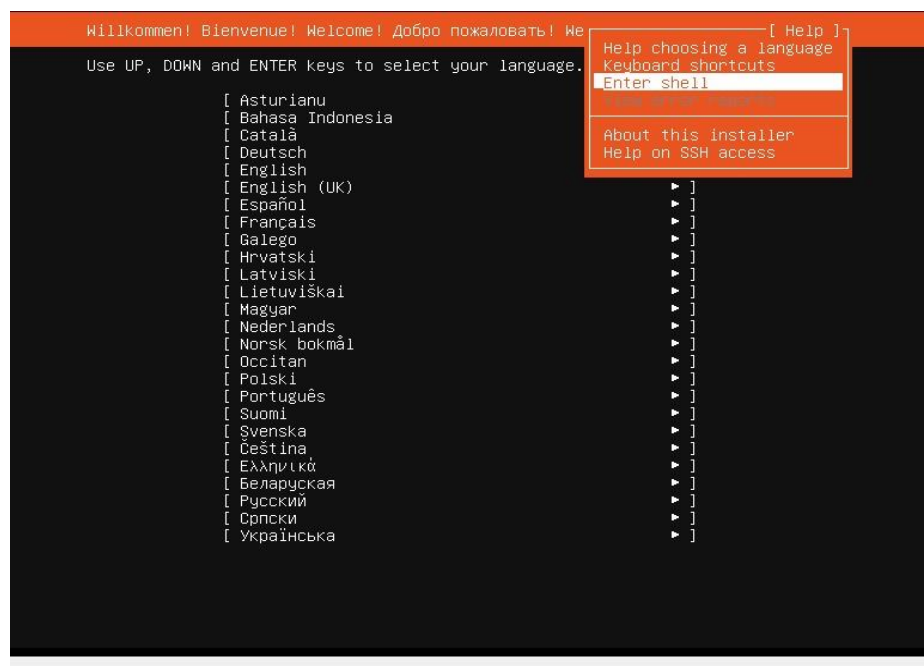
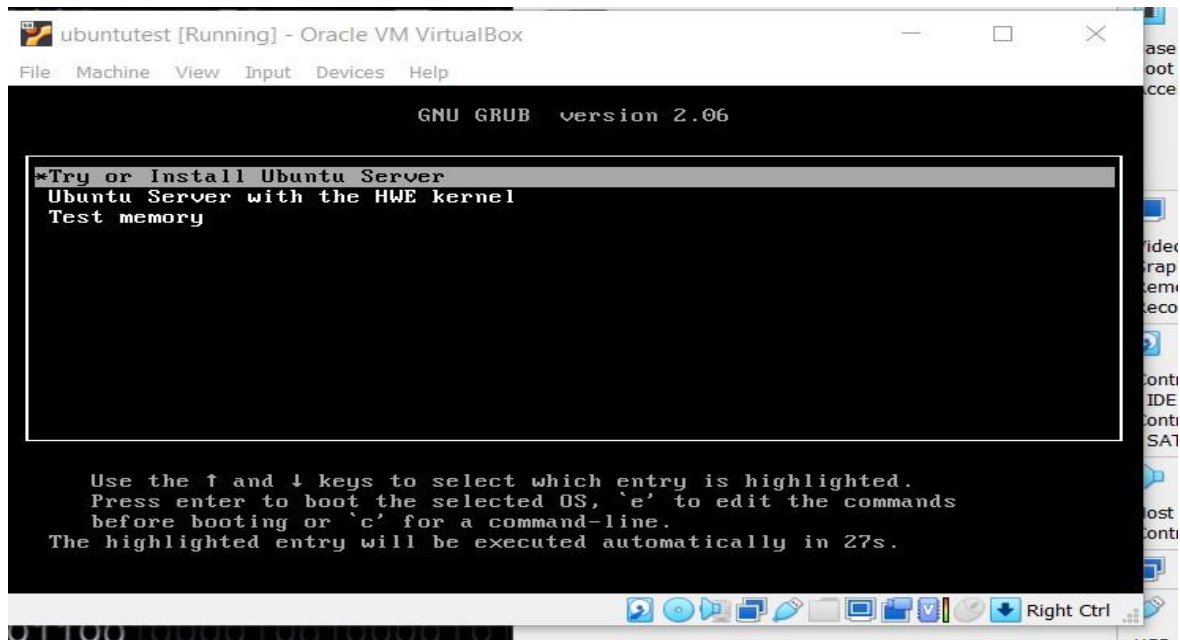
5.3 Finalisation des modifications : Une fois que vous avez terminé vos modifications, sauvegardez votre projet.

5.4 Génération de la nouvelle image ISO.

Guide d'utilisation :

Pour tester l'image iso on va l'installer dans le logiciel "Oracle VM VirtualBox "





Après on accède a "help " après en "entrer Shell" pour tester notre distribution et on a fait cette étape pour entrer à notre distribution directement car si on continue l'opération d'installation il va transformer notre système de fichiers juste un simple fichier qui contient des simples scripts et que ne sont pas intégrer comme des variables d'environnement en Ubuntu.

```
ubuntu-moussa [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Installer shell session activated.

This shell session is running inside the installer environment. You
will be returned to the installer when this shell is exited, for
example by typing Control-D or 'exit'.

Be aware that this is an ephemeral environment. Changes to this
environment will not survive a reboot. If the install has started, the
installed system will be mounted at /target.

ID1 DISTRIBUTION

root@id1distribution:/# whereami
Current directory : /
root@id1distribution:/# cd ..
root@id1distribution:/# ls
ID1FS boot dev home lib32 libx32 var proc root sbin srv tmp var
bin cdrom etc lib lib64 media opt rofs run snap sys usr
root@id1distribution:/# cd /ID1FS
root@id1distribution:/ID1FS# ls
com data design.txt design2.sh etc execut.sh home ko.png permissions.txt recyclebin usr
root@id1distribution:/ID1FS# cd com
root@id1distribution:/ID1FS/com# ls
cat crtfd delete ditusr installpack me rnm signup whereami
cmds crtfd dlthbin inode list openf signin userdata
root@id1distribution:/ID1FS/com# cd ..
root@id1distribution:/ID1FS# cd ..
root@id1distribution:/# crtfd file.txt
File 'file.txt' created successfully with metadata.
root@id1distribution:/# _
```

FIN