



CONSERVATOIRE NATIONAL DES ARTS ET MÉTIERS
CENTRE RÉGIONAL ASSOCIÉ DE NOUVELLE-AQUITAINE

MÉMOIRE

présenté en vue d'obtenir

le DIPLÔME d'INGÉNIEUR CNAM

SPÉCIALITÉ : Informatique

OPTION : Réseaux, Systèmes et Multimédia

par

Marc BENINCA

**Systèmes d'Exploitation autonomes incrémentaux
une inversion des approches conventionnelles**

Soutenu le JJ Mois 2021

JURY

PRÉSIDENT :	Civilité	Prénom	NOM	<i>Fonction</i> <i>Organisme</i>
MEMBRES :	Civilité	Prénom	NOM	<i>Fonction</i> <i>Organisme</i>
	Civilité	Prénom	NOM	<i>Fonction</i> <i>Organisme</i>

Remerciements

Conditions de travail

Abréviations

SE Système d'Exploitation. [6](#)

Glossaire

expression description en français. [6](#)

Plan

Remerciements	2
Abréviations	3
Glossaire	4
Plan	5
Introduction	7
1 Problématique : maintenir des systèmes d'exploitation	8
1.1 Systèmes de fichiers, installés sur partitions, avec accès en écriture	8
1.1.1 Système de fichiers conventionnel	8
1.1.2 Système de fichiers géré par des recettes configuration	9
1.1.3 Système de fichiers avec gestion d'instantanés	9
1.2 Images autonomes, sans installation, avec accès en lecture seule . .	10
1.2.1 Démarrer sans stockage persistant	10
1.2.2 Démarrer avec un stockage persistant	10
1.3 Alternatives existantes	10
1.3.1 Windows Unified Write Filter	10
2 Proposition : un fonctionnement autonome incrémental	11
2.1 Mettre en œuvre le fonctionnement	11
2.1.1 Cloner des dépôts officiels et d'éditeurs tiers	11
2.1.1.1 Synchroniser des miroirs locaux	12
2.1.1.2 Choisir les architectures utiles	12
2.1.1.3 Vérifier l'intégrité	12
2.1.2 Construire un système de fichiers autonome	13
2.1.2.1 Installer les paquets spécifiques	13
2.1.2.2 Créer un système de fichiers minimal	13
2.1.2.3 Équiper un système pour en fabriquer d'autres	13
2.1.2.4 Rendre un système de fichiers autonome	13
2.1.2.5 Installer des paquets supplémentaires	13
2.1.2.6 Lier certaines données à du stockage persistant	13
2.1.3 Encapsuler dans un fichier image	13

2.1.3.1	Utiliser un format adapté au montage en lecture	13
2.1.3.2	Choisir un algorithme de compression	14
2.1.3.3	Encapsuler dans un fichier image hybride	14
2.1.4	Sécuriser un fichier image produit	14
2.1.4.1	Calculer une somme de contrôle d'intégrité	14
2.1.4.2	Signer pour certifier l'authenticité	15
2.1.5	Amorcer des fichiers images sécurisés	15
2.1.5.1	Créer des images de démarrage	15
2.1.5.2	Créer un menu de démarrage	15
2.1.5.3	Vérifier intégrité et authenticité	15
2.1.5.4	Charger des images en mémoire vive	16
2.1.6	Mise à niveau incrémentale	16
2.1.6.1	Fabriquer une nouvelle image	16
2.1.6.2	Éviter un redémarrage facultatif	16
2.1.6.3	Réduire la durée d'un redémarrage obligatoire	16
2.2	Automatiser le fonctionnement	16
2.2.1	Vérifier l'intégrité des dépôts locaux	16
2.2.2	Construire des systèmes de fichiers autonomes complets	16
2.2.3	Créer de nouveaux fichiers par mise à jour d'images	16
2.2.4	Générer un menu de démarrage à la volée	16
3	Résultats	17
	Conclusion	18
A	Annexe	19
	Références	20
	Figures	21
	Tableaux	22
	Résumés	23

Introduction

[Système d'Exploitation \(SE\)](#)

[Expressions](#)

[Debian\[1\]](#)

Chapitre 1

Problématique : maintenir des systèmes d'exploitation

En fonction des cas d'utilisation, maintenir des systèmes d'exploitation mène à penser :

- mises à jour
- indisponibilité
- politique de sauvegardes
- tests des sauvegardes
- instantanés
- restaurations
- recettes de configuration

1.1 Systèmes de fichiers, installés sur partitions, avec accès en écriture

1.1.1 Système de fichiers conventionnel

- ext2
- ext3
- ext4
- jfs

- xfs

Avantages :

- TODO

Inconvénients :

- TODO

1.1.2 Système de fichiers géré par des recettes configuration

- ansible
- chef
- puppet

Avantages :

- TODO

Inconvénients :

- TODO

1.1.3 Système de fichiers avec gestion d'instantanés

- btrfs
- zfs

Avantages :

- TODO

Inconvénients :

- TODO

1.2 Images autonomes, sans installation, avec accès en lecture seule

1.2.1 Démarrer sans stockage persistant

Avantages :

- TODO

Inconvénients :

- TODO

1.2.2 Démarrer avec un stockage persistant

Avantages :

- TODO

Inconvénients :

- TODO

1.3 Alternatives existantes

1.3.1 Windows Unified Write Filter

Chapitre 2

Proposition : un fonctionnement autonome incrémental

Avantages :

- redémarrage = restauration
- mise à jour = sauvegarde
- séparation système et données

Inconvénients :

- maintenance manuelle exhaustive

2.1 Mettre en œuvre le fonctionnement

2.1.1 Cloner des dépôts officiels et d'éditeurs tiers

Avantages :

- TODO

Inconvénients :

- TODO

2.1.1.1 Synchroniser des miroirs locaux

apt-mirror

- traductions (Translation-*.bz2)
- contenus indépendants de l'architecture (Contents-all.gz)
- Contents-*/InRelease avec certains dépôts éditeurs

debmirror

ftpsync

2.1.1.2 Choisir les architectures utiles

amd64

arm64

armhf

i386

2.1.1.3 Vérifier l'intégrité

Avantages :

- éviter des erreurs lors de futures installations de paquets

Inconvénients :

- aucun outil n'existe

2.1.2 Construire un système de fichiers autonome

Debian GNU/Linux

2.1.2.1 Installer les paquets spécifiques

Machine physique

Machine virtuelle

Conteneur

2.1.2.2 Créer un système de fichiers minimal

debootstrap

2.1.2.3 Équiper un système pour en fabriquer d'autres

2.1.2.4 Rendre un système de fichiers autonome

live-boot

update-initramfs

2.1.2.5 Installer des paquets supplémentaires

2.1.2.6 Lier certaines données à du stockage persistant

2.1.3 Encapsuler dans un fichier image

2.1.3.1 Utiliser un format adapté au montage en lecture

SquashFS

2.1.3.2 Choisir un algorithme de compression

gzip

lzma

lzo

lz4

xz

zstd

2.1.3.3 Encapsuler dans un fichier image hybride

ISO

2.1.4 Sécuriser un fichier image produit

2.1.4.1 Calculer une somme de contrôle d'intégrité

SHA-256

SHA-512

2.1.4.2 Signer pour certifier l'authenticité

2.1.5 Amorcer des fichiers images sécurisés

2.1.5.1 Créer des images de démarrage

GRUB

BIOS

UEFI

Secure boot

2.1.5.2 Créer un menu de démarrage

grub.cfg

loopback

squash4

iso9660

2.1.5.3 Vérifier intégrité et authenticté

gcry_sha256

gcry_sha512

pgp

2.1.5.4 Charger des images en mémoire vive

overlayfs

2.1.6 Mise à niveau incrémentale

2.1.6.1 Fabriquer une nouvelle image

2.1.6.2 Éviter un redémarrage facultatif

Avantages :

- aucune indisponibilité
- répéter simplement les modifications sur le système en mémoire

Inconvénients :

- TODO

2.1.6.3 Réduire la durée d'un redémarrage obligatoire

kexec-tools

2.2 Automatiser le fonctionnement

2.2.1 Vérifier l'intégrité des dépôts locaux

2.2.2 Construire des systèmes de fichiers autonomes complets

2.2.3 Créer de nouveaux fichiers par mise à jour d'images

2.2.4 Générer un menu de démarrage à la volée

Chapitre 3

Résultats

Conclusion

Annexe A

Annexe

Références

- [1] Raphaël Hertzog et Roland Mas. *The Debian Administrator's Handbook*. Buster. 2020. url : <https://debian-handbook.info>.

Figures

Tableaux



Systèmes d'Exploitation autonomes incrémentaux une inversion des approches conventionnelles

Mémoire d'Ingénieur CNAM,
Bordeaux 2021.

RÉSUMÉ

Ligne 1.
Ligne 2.
Ligne 3.

Ligne 4.
Ligne 5.
Ligne 6...

**Mots clés : un, deux, trois, quatre,
cinq, six, sept, huit.**

SUMMARY

Line 1.
Line 2.
Line 3.

Line 4.
Line 5.
Line 6...

**Key words: one, two, three, four,
five, six, seven, eight.**