

# Débuter en informatique selon la philosophie arch

Auteur: Anthony JR. Le Goff

Date: Novembre 2021

Licence:

Copyright (C) 2021 ANTHONY JR. LE GOFF.

Permission is granted to copy, distribute and/or modify this document under the terms of the **GNU Free Documentation License, Version 1.3** or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled “[GNU Free Documentation License](#)”.

## 1. Pourquoi ce livre?

On peut se demander à quoi cela sert d’écrire. Je dirais pour transmettre du savoir. Mais on peut élaborer une pensée, et intellectualiser à l’écrit pour libérer des concepts et forcer à la réflexion. Ce livre n’est pas encore un livre pour débuter en informatique sans une réflexion globale d’un éco-système, son fonctionnement et son architecture en traduisant un cas pratique. C’est un cadre éducatif pour la vulgarisation de l’informatique.

Mon sentiment est de partager ce que j’ai appris et les bonnes pratiques techniques de l’utilisation du système sous GNU/Linux. Ce qui équivaut à des heures de travail à rechercher de la documentation, regrouper des sources d’information et analyser pour une implémentation. Ce livre est donc une synthèse de mes connaissances pour établir une forme de *handbook*.

Ce livre s’adresse à tous les curieux, qui ont entendu parler de *Linux* et voudrait apprendre l’informatique par la pratique. J’y fait une approche *hacker* que je vais m’efforcer de définir. Ainsi ce livre se veut d’être un indispensable pour débuter dans le hacking avec les meilleurs outils mis à la disposition. L’intérêt est de faciliter la mémorisation de nombreux concept informatique et d’accélérer la courbe d’apprentissage. A la fin de ce livre, l’utilisateur aura clé en main un système à jour selon les dernières technologies du marché pour:

- Faire de la recherche en sécurité informatique
- Développer et concevoir des applications de logiciel libre
- S’initier au *hack* dans son ensemble avec le matériel libre
- Collaborer, s’initier à des projets open source
- S’informer et collecter de l’information
- Protéger son système de l’intrusion, corruption de fichier et au droit à la vie privée

En clair donner les meilleurs outils pour appartenir à une communauté, rendre accessible l'informatique. Je ne pouvais pas élaborer une page de ligne de commande, car je veux que le lecteur commence par étudier la ligne de commande en expliquant dans le détail. On n'apprend pas l'informatique en restant sur une interface graphique, c'est une aide qui occulte tous les principes sous-adjacent.

L'idée vient du mot anglais *from scratch* ou une façon de dire à construire soi-même. Peut-être que vous avez eu des rumeurs d'utilisation de bureau ultra-personnalisable sous GNU/Linux mais que vous avez aucune idée comment vous y prendre même si l'idée de customiser le système selon vos besoins vous attire.

Ce livre appartient à la philosophie de la sous-culture libre. Il est donc accessible selon la licence défini. Le logiciel utilisé sous GNU/Linux pour son écriture est Ghostwriter pour l'édition en Markdown. C'est le format de transition facilitant l'exportation en HTML mais également en PDF ou EPUB. Je recherche à publier une version en ligne via des outils open source tel que Gitbook. Ce livre se veut interactif avec des liens internet pour approfondir un sujet ou l'utilisation de référence d'ou le format utilisé. L'utilisation de Git devient un standard dans le milieu informatique que je vais développer. Dans ce sens je veux mettre à disposition mes écrits que cela soit pour la réflexion et la pratique de l'informatique dans la culture libre. C'est un travail qui devrait permettre de convertir de nouveaux aspirants *hacker* selon la tradition depuis les débuts de l'informatique et une philosophie de libre partage de l'information.

J'espère bien gagner en influence et réputation pour devenir une référence de formation aux outils de hacking du débutant en informatique. Cela me convient plus que faire de l'argent sur le livre, la connaissance doit être libre d'accès en particulier les outils de création informatique dans l'optique de produire du savoir, ajouter de la valeur et innover pour faire progresser la communauté.

Publier le livre chez un éditeur commercial aurait été contre-productif car je restreins le savoir et limite la diffusion à un support. La plupart des pirates informatiques ne payent pas de livre, dans le milieu il existe des moyens de trouver des PDF ou de la documentation en ligne gratuitement et par des torrents.

## 2. Préambule libriste

Il m'était impossible de ne pas introduire l'objet de l'étude de l'informatique par un mouvement de fond tel que le logiciel libre. Steve Ballmer(ex PDG Microsoft) disait du logiciel libre que c'était un cancer pour la propriété intellectuelle. Ce que l'on entend par propriété intellectuelle est le brevet pour une invention. Ce qui permet de rémunérer les inventeurs et d'accéder à la paternité d'une nouvelle idée mise en concept à bus de faire du profit commercial. C'est donc une manière d'éviter de ce faire copier et donc voler son travail. Personne n'aime le plagiat, qui représente un travail médiocre en qualité. Dans ce sens l'arrivée des logiciels informatiques allaient remettre en cause les pratiques dominantes sur les oeuvres de l'esprit. En particulier l'étude et la redistribution du code source. Un logiciel disposant du code source peut-être copié à l'infini ne vous dépouillant pas de votre travail ayant vous même une copie du code source. L'étude du code source

permet donc d'améliorer les fonctionnalités et d'en faire des dérivées *fork*, de l'adapter sur plusieurs matériels mais également d'en faire un travail d'audit de recherche de faille de sécurité qui pourrait compromettre son utilisation. Ces quelques principes sur une économie du partage n'est pas venu instinctivement car des pratiques pour s'enrichir sur le dos du modèle capitaliste vont apparaître fin des années 70. Le code ouvert en particulier par l'utilisation des universitaires se fermera par l'arrivée commerciale pour le grand public du micro-ordinateur dont le système d'exploitation DOS. Quelques projets vont rentrer en résistance dont le projet GNU de Richard Stallman en mémoire de l'esprit hacker des premiers jours pour faire un dérivée de système d'exploitation UNIX qui serait libre. On peut dire que Richard Stallman est le premier libriste en définissant le logiciel libre. Une véritable bataille du libre va donc commencer marquant l'histoire de l'informatique.



GNU Logo

Dans ce cadre, la volonté de la communauté est avant tout chose de résoudre un problème technique que de répondre à une offre marchande et donc la protection par un brevet contre la concurrence. Pourtant il est nécessaire de considérer la paternité d'une innovation technologique en informatique et de prendre en compte les auteurs. Ce que va définir Richard Stallman dans son projet GNU est une licence d'utilisation. Les libristes donc se définissent comme des enfants du projet GNU et la définition des quatre libertés fondamentales tel que:

- la liberté d'exécuter le programme, pour tous les usages ;
- la liberté d'étudier le fonctionnement du programme et de l'adapter à ses besoins ;
- la liberté de redistribuer des copies du programme (ce qui implique la possibilité aussi bien de donner que de vendre des copies) ;
- la liberté d'améliorer le programme et de distribuer ces améliorations au public, pour en faire profiter toute la communauté.

Ainsi la nécessité d'accès au code source est une condition de ces libertés. Et après tout quand on y réfléchit bien n'est-ce pas pareil en biologie? Un être humain a la liberté d'être copié et de se reproduire et son programme codé dans l'ADN est disponible. Aujourd'hui en génétique on peut modifier cet ADN pour l'améliorer ou réparer des caractéristiques. On peut penser au processus d'amélioration continue et de mutation d'un logiciel comme l'évolution du vivant. Avec des branches dans l'origine d'un arbre d'une espèce. Ainsi un logiciel pour le traitement de texte est une famille qui demande d'être adapté à un environnement de travail qui évolue et donc mise à jour dans le temps en ajoutant des caractéristiques. Certaines personnes vont ajouter des

fonctionnalités exclusives pour en tirer parti d'un point de vue commerciale. Tel est le constat caché le code source complique le processus d'innovation car on restreint la distribution à quelques acteurs et donc on élève des privilèges. Cela me rappelle des affaires tel que Coca-Cola qui garda jalousement secrète sa recette et donc garder un monopole. Hors c'est bien la lutte contre les monopoles dont il est question dans le logiciel libre. C'est une stratégie qui est contre-productive car sur du long terme vous ne pouvez pas empêcher la recherche, également cela engendre dans le principe de concurrence une mise en place d'espionnage pour découvrir les secrets de fabrication et de production. Si on veut réduire la malveillance un secret doit être partagé. On retrouve dans d'autres domaines des exemples de raté, tel que la fabrication de la bombe atomique. les USA n'ont jamais gardé le monopole, au contraire cela n'a fait qu'accélérer une course à l'armement. La question de la propriété intellectuelle n'est pas vite répondu, la découverte d'une solution technique engendre un phénomène d'adoption rapide à partir du moment qu'il répond à un besoin. Dans le cadre de la bombe atomique, c'est la question de puissance qui est remis en cause et donc l'impact politique. Mais cela peut-être un impact sur la création tel que des logiciels de CAO(conception assisté par ordinateur). Doit-on restreindre la possibilité de création à l'industrie et quelques licences universitaires? Comment fait-on pour que l'étudiant ce forme si celui-ci ne peut qu'utiliser les ordinateurs à l'université? Et si on donne un travail personnel à faire chez soi? Que de contrainte. Au-delà de ça c'est croire une utopie de civilisation que la recherche de solution technique ne se fait qu'à l'industrie dans le cadre d'un travail rémunérer. Hors, l'histoire des découvertes de ce monde en science est une affaire de passionné et pas seulement qu'en laboratoire. C'est ce que démontre le logiciel libre par le bénévolat et hors temps de travail, on peut produire une solution technique. Il pourrait être question de développement durable et d'accès à la technologie pour des pays en voie de développement. Pour accéder à ces marchés le coût pour se fournir en matériel informatique est bien trop élevé. Donc on regarde des solutions alternatives parfois gratuite pour remplacer les industriels et logiciels privés. La circulation de l'information doit être accélérée en tout point du globe et définir les bonnes pratiques pour que chacun puisse avoir accès aux outils de création, de production en particulier d'oeuvre de l'esprit et donc intellectuel.

C'est sous cette nécessité que ce livre se construit. Par adhésion au mouvement de la culture libre initié par des informaticiens pour aller plus loin que l'idéologie, c'est un projet de société et de civilisation et la croyance en une futur économie de l'abondance dont le logiciel libre est la première brique qui déposa un standard sur l'utilisation de ressource illimité. Elle doit être tout d'abord développé dans un paradigme de la pensée et de l'immatériel dont la révolution informatique à propager l'économie de la connaissance actuelle, vers la possibilité pour l'humanité d'élargir les horizons vers l'espace et le minage d'astéroïde et de exoplanète permettant l'utilisation de la ressource matériel illimité réduisant les conflits sur les problématiques de rareté. Il faut être clair pour tout programme spatial, c'est tout d'abord du calcul de trajectoire de projectile que l'informatique nous permet et la production d'un capital immatériel pour développer un produit avec succès. Le logiciel libre est le cheval de Troie de l'ouverture vers l'espace car c'est un univers fait de robotique terrain de jeu de la cybernétique. Aujourd'hui du

logiciel libre est embarqué dans les fusées SpaceX ou sur le rover Persévérance sur Mars. Le succès du logiciel libre à travers la FSF(Free-software foundation) a permis de voir des projets tel que:

- Linux (noyau de système d'exploitation)
- Apache (serveur web)
- Mozilla Firefox (navigateur internet)
- Gimp (logiciel de dessin)
- Git (outil de contrôle de version)
- Perl (langage de programmation)

Largement adopté par les industriels de nos jours par les pratiques des développeurs car répondant une solution technique et un réel besoin. 99% top 1 million des websites fonctionnent sur un serveur GNU/Linux et 100% des top 500 supercalculateurs mondiaux ont adopté GNU/Linux. Il y a une véritable suprématie des logiciels libres quand il est question de performance et de réduire les coûts de fonctionnement. On peut également noter que les scientifiques du CERN tournent sous GNU/Linux et sont très actifs dans le développement par la propagation de la science ouverte. Le logiciel libre est moteur dans la mise en oeuvre de pratique développée par l'économie du partage qui réduit de plus en plus la portée des documents classifiés chez les industriels dans une optique de collaboration entre des parties prenantes pour chasser en meute.

### 3. L'héritage d'UNIX

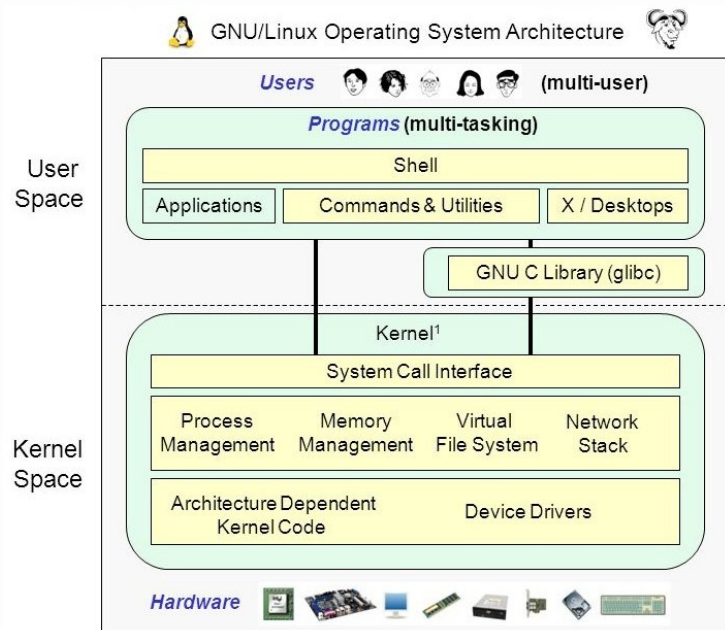
Revenons un peu sur la genèse du projet GNU, j'expliquai que ce livre est de vous permettre d'apprendre l'informatique et d'acquérir des libertés numériques. Nous allons donc faire une étude de cas à travers une distribution GNU/Linux. Il faudra donc connaître GNU et Linux. Comme j'en parlais le projet GNU est une ré-écriture complète d'UNIX. UNIX est un système d'exploitation pour ordinateur, c'est à dire que c'est un logiciel qui permet la communication entre le matériel(hardware) physique tel que le clavier, l'écran, la souris, le processeur, etc... et des applications comme votre navigateur internet. GNU est une série de programmes, bibliothèques et compilateurs pour l'utilisateur: c'est à dire un environnement complet de travail pour le développeur pour créer des logiciels. Vous trouverez à la page [GNU du wiki arch linux](#) la liste complète de projet GNU de nos jours. Au commencement, en 1985, on retrouvait en particulier:

- Bash, un shell pour écrire des lignes de commande
- glibc, *GNU C Library* Implémentation de la bibliothèque standard en C
- Emacs, un éditeur de texte
- gcc, GNU Compiler Collection est un logiciel de compilation en C

UNIX a été écrit en langage C. Les systèmes d'exploitation sont écrits dans ce langage, il était donc nécessaire de créer un compilateur pour faire des fichiers binaires exécutables. Le compilateur transforme du code que peut lire et écrire l'humain en langage C vers du code machine binaire à base de 0,1 des bits que peut lire le processeur et exécuter. Pour écrire du code, on utilise un éditeur de texte. Sous UNIX, l'éditeur est Vi. Il existe une petite guerre de religion entre les puristes qui utilisent Emacs et Vi puis son héritier Vim. Mais cela vous va-t-il aider à découvrir. On utilise un shell pour lancer un programme tel que Vi en ligne de

commande pour éditer un fichier en langage C puis on compile toujours en ligne de commande avec gcc. Quand à glibc est une bibliothèque C qui apporte les appels système et les fonctions de base telles que open, malloc, printf, etc. Si vous cherchez à étudier la programmation système il vous faudra connaître ces notions, mais pour l'instant ce n'est que pour la culture.

## 4. Linux



Architecture OS GNU/LINUX

Le projet GNU n'est pas complet, il manque un noyau pour dialoguer avec le matériel de l'ordinateur, le noyau Hurd sera une tentative de concevoir une solution, mais qui sera pas opérationnel. Cette partie sera introduite par un finlandais du nom de Linus Torvalds en 1991 comme bénévole. Celui-ci va écrire un noyau à l'origine avec 10 000 lignes de code d'après le projet Minix. Le noyau Linux fonctionne pour les architectures de processeur x86 et fait 64Ko une fois compressé. La version 0.0.1, disponible sur le visualisateur de [code source du noyau](#), était composé d'une cinquantaine d'appel système, c'est à dire la gestion de base pour les processus, la mémoire et les fichiers. Ainsi le système d'exploitation sera finalisé par une fusion des projets GNU/Linux et l'arrivée d'interface graphique(GUI) comme le projet Debian. Une entreprise va rapidement entrée dans le jeu avec la fondation de Red Hat par Robert Young. C'est la naissance d'un éco-système économique commercial et communautaire. En 2021 il existe environ 15 600 développeurs du noyau qui ont contribué depuis 2005 et environ 1400 compagnies. Grâce à l'arrivée de Git il est possible de tracer les contributions. Il est possible de devenir un "kernel dev" et de hacker le noyau. La [documentation](#) pour contribuer pour le noyau est copieuse sur internet ainsi que des formations tel que chez la "Linux Foundation" gratuitement. Il faudra tout d'abord connaître le langage C comme prérequis et être à l'aise avec Git.

## 5. Eco-système

On l'aura vue, depuis la genèse du projet GNU de Richard Stallman, un écosystème va se mettre en place avec des programmeurs autour du logiciel libre qui donnera naissance à Linux. Tout d'abord la FSF ou free-software foundation. On va retrouver le premier système d'exploitation sous GNU/Linux et la communauté Debian, puis viendra la première entreprise avec Red Hat puis la distribution Fedora. On compte maintenant plus de 600 distributions sous GNU/Linux dans le monde avec des partisans et militants du logiciel libre. En France la promotion et la défense du logiciel libre est réalisée par l'April, mais aussi par l'association Framasoft qui publie des documents et ressources et également Linuxfr.org qui est le principal site communautaire. En 2021 la documentation la plus accessible est celle de la distribution Ubuntu pour s'initier à la communauté ainsi que le wiki Arch Linux sur internet. A noter que Wikipédia, l'encyclopédie libre est une réussite du mouvement, autant apprécié des internautes curieux de connaissance que des thésards comme source d'information. Les idées reçoivent comme quoi on ne peut pas faire confiance à une source d'information que n'importe qui peut éditer et contribuer n'est plus d'actualité grâce au travail des wikipédiens.

Un autre mouvement va apparaître en 1998 sous l'impulsion de Eric Raymond nommé l'OSI (Open source initiative) mettait plutôt l'accent sur le développement technologique en permettant aux créateurs de se servir librement du travail de leurs devanciers. L'open source sera particulièrement plus adapté pour les entreprises pour développer de la technologie alors que le logiciel libre sera plus un mouvement social selon Richard Stallman. Lawrence Lessig, le créateur de Creative Commons (CC) et du livre "culture libre" en 2005 fera la remarque que le copyright est un obstacle à la production culturelle, au partage de la connaissance, à l'innovation technologique et à l'intérêt privé. Une autre ONG représente le paysage de la culture libre à l'échelle internationale est l'EFF (Electronic frontier foundation). L'EFF contribue à la protection des droits et libertés sur internet. L'objectif essentiel de l'EFF est de défendre la liberté d'expression sur Internet, et plus largement la vie privée en ligne des utilisateurs. Pour accompagner ces derniers dans cette optique, EFF a notamment mis en place un guide spécialisé dans la protection contre l'espionnage en ligne, intitulé Autodéfense contre la surveillance.

## **6. Les hackers**

## **7. Ou trouver de l'aide?**

<https://www.webminal.org/>

## **8. Keep it simple, stupid**

## **9. La modularité**

## **10. Le matériel**

**11. Obtenir la dernière version d'Arch Linux**

**12. Booter sur une clé USB**

**13. Débuter l'installation**

**14. Connectez-vous au réseau internet**

**15. Sélectionner la source**

**16. Configurer l'horloge**

**17. Partitionner le disque**

**18. Chiffrer le disque**