PRESENTATION - LINUX / OS

Système, vie privée, outils alernatifs

Presentation de Linux

- Linux utilise un noyau/kernel Unix libre
- Unix est présent dans de nombreux systèmes (FreeBSD, OpenBSD, OSX)
- Coeur du système Linux (gestion mémoire, pilotes, processus)
- Fournit plusieurs API et sert au fonctionnement réseau
- Utilise en parallèle GNU (Ancien OS, intégrant pas mal de tooling)
- Le Kernel avec GNU forment le système d'exploitation final
- Les distributions ajoutent des fonctionnalités supplémentaires

L’indispensable Systemd

- Fournit une gestion des dépendances entre services
- Permet le chargement des services au démarrage
- Ajoute une meilleure parallélisation
- Présence des fichiers XDG (utilisés pour les DE)
- Journalisation et présences de logs
- Certains OS (ArtixLinux) sont systemd free et plus complexes

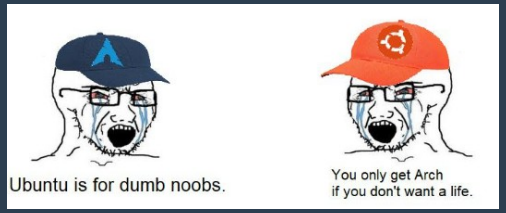
Bootloader

- Stocké dans le MBR et permet de charger le système en mémoire
- Grub est le plus connu et a longtemps été utilisé (multiboot, shell, net)
- Est de plus en plus remplacé par systemd-boot
- Systemd est plus robuste et directement intégré
- Il reste malheureusement moins flexible pour le multiboot
- Des failles ont étés découvertes, avec Grub2 (BootHole).
- Une alteration permettait d'insérer du code malveillant et de le charger en mémoire lors du boot

Sans rien, c’est à la mano !

- Sans ajout, vous n’avez accès qu’à du texte (avec le root shell)
- Implique de séparer les droits et d’avoir autre chose que le root
- Souvent le groupe sudoers est utilisé
- Sudo permet d’exécuter des commandes « superutilisateur » sans avoir à se connecter directement avec le compte root.
- Vous devez installer chaque composant, firewall, pilotes, interfaces, gestionnaire de fichier, visionneuse d’image, navigateur
- Il est possible de le faire via makepkg ou bien via un gestionnaire
- Installer un DE permet d’obtenir une suite d’outils (interface, fenetres…)

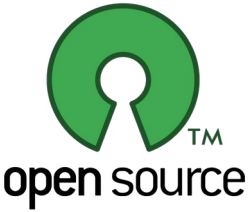
Top des distributions GNU/LINUX



- EndeavourOS (Arch) – (minimal, nécéssite de bidouiller, dépots AUR, binaires/compil)
- Manjaro (Arch) – (Plus d'outils, user friendly, dépots AUR, binaires/compil)
- Unbuntu (Debian) – (Simple pour débuter, bon DE, dépots corrects)
- Mint (Debian) – (Simple pour débuter, bon DE, dépots corrects)
- PopOS (Debian) – (Base débian, intermédiaire, super windows manager et DE)
- NixOS (Nix) – (Packet manager unique, approche DevOps, nombreux outils)
- Gentoo/Funtoo – (Hard, nécéssite de compiler, peu de binaires)
- OpenSUSE – (Ultra stable, fonctionnement conservateur, beaucoup d'outils)
- Fedora – (Packet manager unique, supporté par RedHat, adapté aux devs)

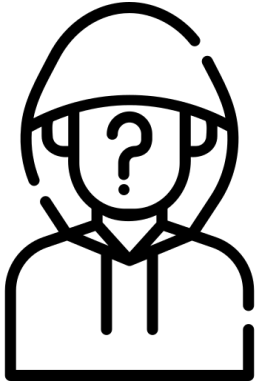
Pourquoi l’utiliser ?

- Le choix, c’est cool, encore plus si c’est gratuit
- Avec juste ce qu’il vous faut, vous réduisez la surface d’attaque
- Mais aussi l’ensemble des bugs possibles et allégez le système total
- Possibilité d’avoir une ergonomie accrue (DE, windows manager)
- Vous pouvez tout automatiser et répondre à vos besoins
- Pas de télémétrie, sauf si vous utilisez du propriétaire (ou des trucs zbull)
- Vous avez le choix entre stabilité ou dernières technos
- La communauté est ultra active, c’est 3 %, mais tous des barbus énervés
- Un bug ou autre, il sera résolu dans l’heure selon votre distribution



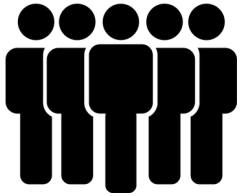
Vie privée / protection

- Vous avez le contrôle total sur votre système
- Vous choissez ce que vous voulez mettre à jour ou non
- La surface d'attaque est clairement réduite
- Le choix d'outils libres est un plus, surtout ceux axés sécurité
- Votre système ne dépend pas des GAFAM ou BATX
- Moins vous avez de softwares, moins vous êtes vulnérables
- La communauté est très active et donc votre système est rapidement patché
- Le code est dispo, avec des tonnes de solutions de sécurisation mais aussi de contournement



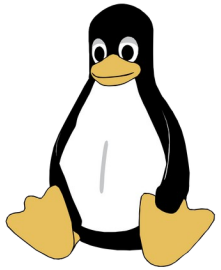
Forum / communauté

- Vous avez des tonnes de forums de passionnés, tous techniques
- Wiki et forum Arch fournit des tonnes d'informations
- Les systèmes basés sous débian ont tous le même fonctionnement
- Gentoo/Funtoo, un Discord est présent, mais laissez pousser votre barbe
- Vous voulez une feature, un mec l'a déjà fait, toujours…
- C'est pas compatible, un type l'a rendu compatible, sachez le !

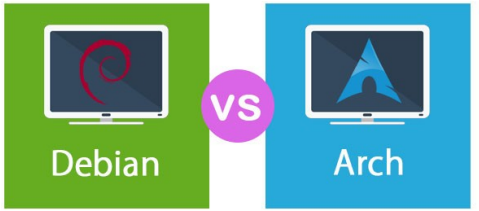


Open source vs propriétaire

- L’open source est flexible et profite des dernières innovations
- La maintenance est communautaire et rapide
- De nouvelles features ou des fork peuvent être crées très rapidement
- Pas toujours de support, mais des forums et de l’entraide générale
- Le propriétaire est maintenu par le fournisseur de logiciel
- Vous acceptez de mettre votre confiance dans un unique organisme
- En revanche vous bénéficiez des services « Professionnels »
- Des contrats peuvent garantir un niveau de service ou d’opérationnalité
- Un support est souvent présent et viendra résoudre vos problématiques



Stabilité vs rolling release



- Les système comme débian sont dit « Stables »
- ArchLinux est plus décrit comme « rolling release »
- Un système stable vous assure de ne pas avoir de bug majeur
- Mais vous oblige à rester avec des versions antérieures des logiciels
- Chaque élément est testé et approuvé, pour ensuite être déployé via une grosse maj
- Arch possède des dépots AUR centralisés, approuvés par la communauté
- Chaque jour des nouvelles versions sont déployés et peuvent affecter le système
- Vous bénéficiez en revanche des dernières maj softwares et pilotes
- Le choix doit se faire en fonction de votre utilisation et niveau d'investissement

Virtualisation / compatibilités

- Contrairement aux idées recues, il est possible de faire tourner des .exe
- Wine est désormais très évolué et peut faire tourner des logiciels MS (Office...)
- Pour les jeux, Steam a publié « Proton » qui fait tourner 98 % des jeux en .exe

- Il est possible de faire de la virtualisation d'un OS Windows
- Les hypervieurs utilisant le kernel fournissent des performances « Desktop »
- Il est possible de configurer du « passthrough » sur vos périphériques

- A titre d'exemple, vous avez KVM ou bien QUEMU
- Vous avez le meilleur du monde opensource et la possibilité de run du propriétaire

Protection et sécurisation

- La gestion des privileges « utilisateurs » est meilleure que sous Windows
- Vous avez des chiffrements natifs variés (LUKS, Tomb…)
- Support facile du système de fichier ZFS (évite les erreurs d'écritures)
- Pas de registre et les binaires sont différents (vous évitez la plupart des malwares)
- Juste ce qu'il faut de services, de ports ouverts, la surface d'attaque est minime
- N'est quasiment jamais déprécié, les mises à jour continuent toujours
- Vous représentez 3 % du parc, les phishing avec code malveillant, c'est pas pour vous
- Pas de traçage, télémétrie, avec un VPN, TOR et un browser correct, vous êtes safe
- Si vous êtes noob, des scripts sont dispo pour automatiser la sécurisation OS

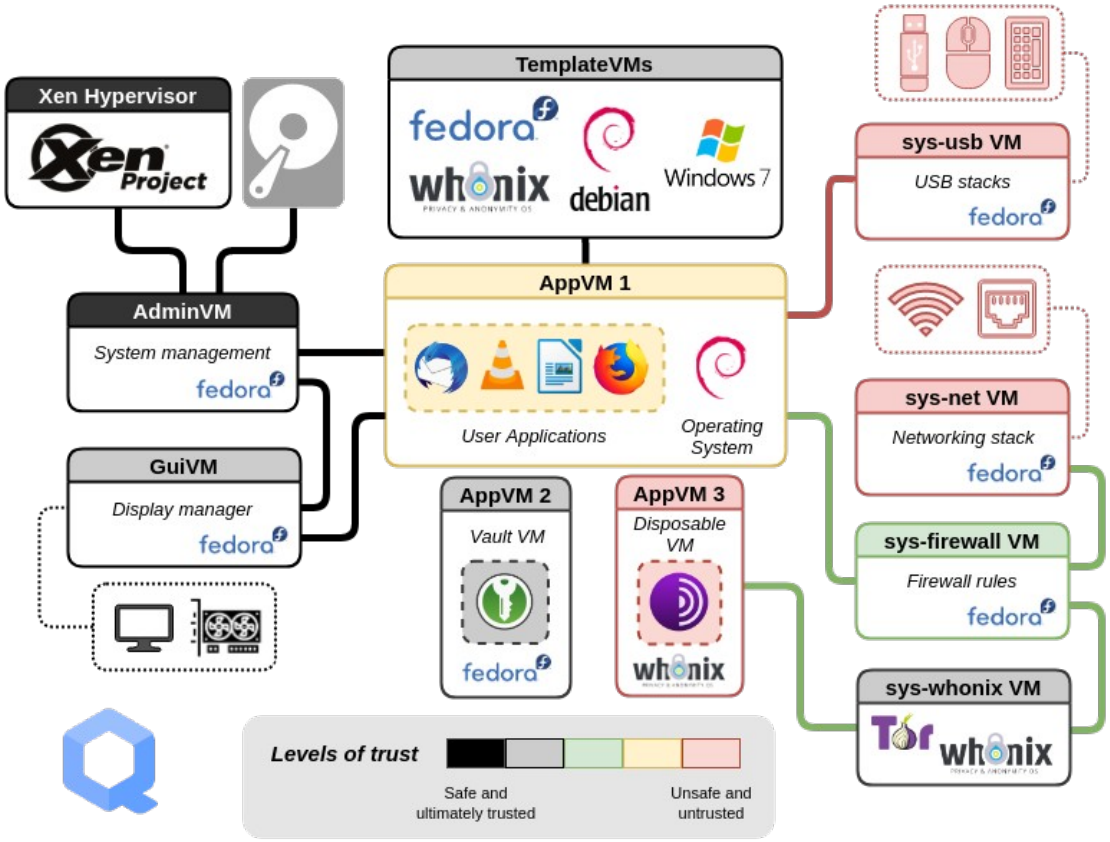
Anonymat : Les indispensables

- On utilise pas Google, remplasez par DuckDuckGo ou Quant
- Navigateur Firefox, avec paramétrage de sécurité (blockage trackers, sites, downloads)
- Des forks sont présents, exemple LibreWolf
- Si vous adorez Chrome, vous avez Chromium « Degoogled »

- On passe par un VPN no log (Ex : Mullvad) avec killswitch en cas de soucis
- On évite d'utiliser des services chez les grand hébergeur (préférez l'autogéré)
- On chiffre tout ce qui est possible, toujours et on évite de stocker la clé en local (lol)
- Pourquoi pas utiliser TOR par dessus, votre connexion passera par des relais
- Pour la messagerie/instant, choisissez bien vos services, ne dévoilez jamais d'infos

KALI LINUX – Le pentesting tout en un

QUBES OS – Isolation maximale des composants

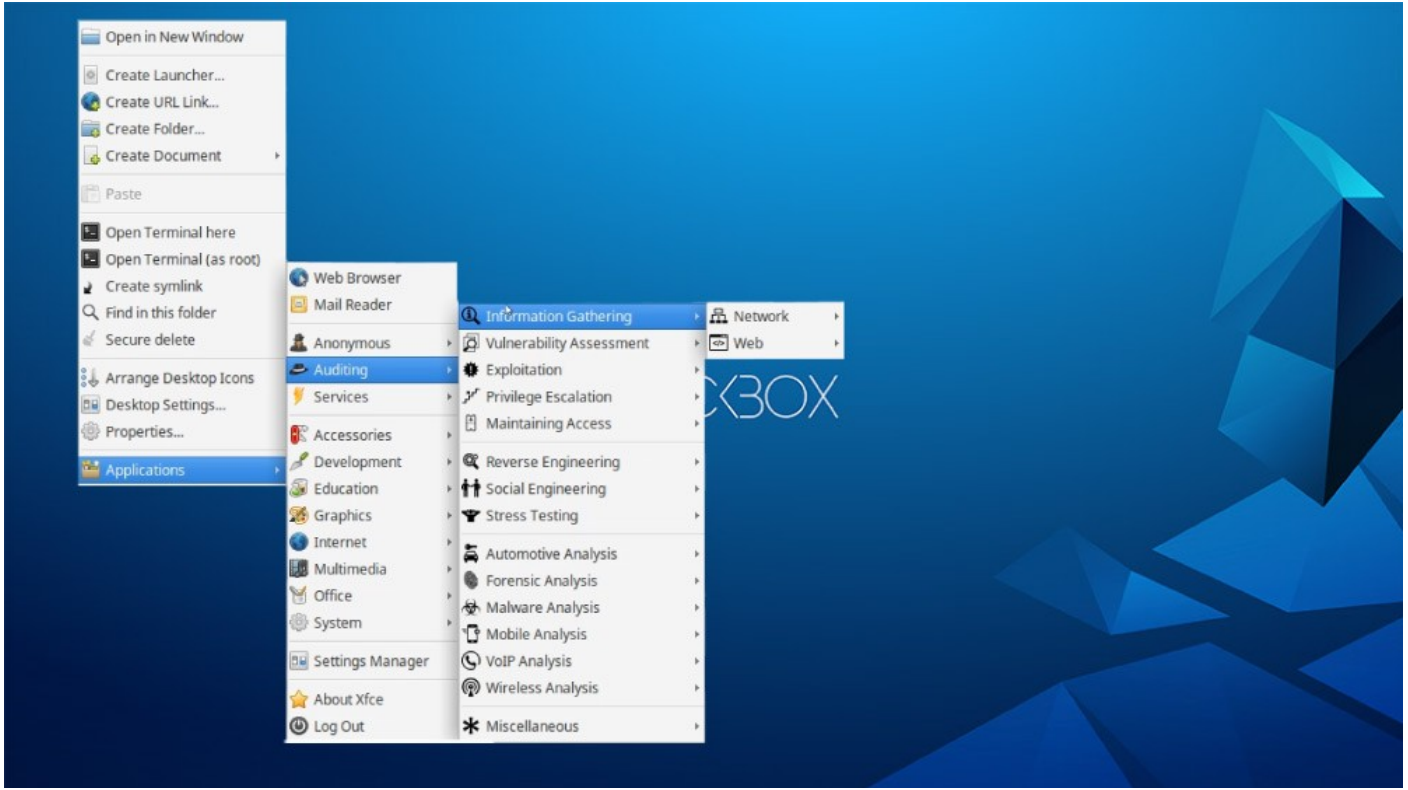


TAILS – Tous anonymes, pas de traces

BLACKARCH – Arch pour les pentesters

PENTOO – Les barbus compilateurs du pentest

BACKBOX – Unbuntu avec des pentest tools



PARROTSEC – Encore du Debian pour pentesters

Whonix – Une Gateway réseau et un OS complet