

Stratégie de sécurité grâce au logiciel libre

Frédéric Raynal <pappy@miscmag.com>
Cédric Blancher <blancher@cartel-securite.fr>



- Introduction
- Le logiciel libre et la sécurité
- GNU/Linux et la sécurité

Pause

- La sécurité grâce au logiciel libre
- Les outils libres de sécurité



Introduction

Qu'est-ce qu'un logiciel libre ?

- Un logiciel utilisable sans restriction
- Un logiciel distribuable sans restriction
- Un logiciel dont on peut étudier le code source
- Un logiciel qu'on peut modifier pour l'adapter à ses besoins, et redistribuer ses modifications

Quatre libertés fondamentales qui définissent le logiciel libre

Ces libertés sont garanties par des licences de distributions :

- GNU GPL
- BSD
- Apache
- Artistic
- etc.



Le noyau Linux et la majorité des bibliothèques et logiciels associés sont distribués sous GNU GPL

Cette licence comporte une clause “contaminante”

- Une modification d'un logiciel distribué sous GNU GPL doit être redistribuée sous GNU GPL
- “Libre un jour, libre toujours”

Les conséquences de l'application d'une licence libre sont multiples

En particulier, nous allons considérer ces conséquences dans le monde de la sécurité informatique



Le logiciel libre et la sécurité



Le besoin de sécurité est lié à un manque de confiance

L'existence de risques divers liés à l'utilisation d'un système d'information introduit un manque de confiance

Ce manque de confiance nécessite le déploiement d'outils propices au rétablissement de la confiance



Quels sont les champs d'application de cette confiance ?

- Confidentialité
- Intégrité
- Disponibilité
- Probité



Confidentialité

- L'information ne doit être accessible que par ce qui est habilité à y accéder



Intégrité

- Les informations stockées, traitées et/ou transportées par un système d'information ne doivent pas être modifiées de manière non prévue



Disponibilité

- Le système d'information doit être en mesure de fournir ses services dans un temps de réponse borné



Probité

- Les actions menées sur un système d'information doivent pouvoir être prouvées



Nécessité de trouver des logiciels capables de remplir ces quatres conditions

- Pour se faire, nous devons être capable de nous assurer qu'ils le font comme nous l'attendons



Pour bien remplir sa tâche, un logiciel de sécurité doit fournir plusieurs garanties

- Transparence
- Adaptabilité
- Pérennité de fonctionnement

Le logiciel libre fournit ces garanties par le fondement que sont les quatres libertés

- L'accès au code source contribue à la transparence
- La liberté d'utilisation, l'accès au code source et le droit de modification assurent l'adaptabilité
- L'accès au code source, le droit de modification et de redistribution assurent la pérennité



Conclusion :

- Le logiciel libre permet de satisfaire les besoins de sécurité d'un système d'information
- Le logiciel libre fournit des outils spécialisés pour mettre en place la sécurité du système d'information (i.e. outils de sécurité)



GNU/Linux est un système d'exploitation libre parmi d'autres :

- GNU/Linux
- OpenBSD
- NetBSD
- FreeBSD

GNU/Linux est le plus répandu d'entre eux



GNU/Linux est un système d'exploitation composé de :

- un noyau Linux (GPL)
 - des bibliothèques et outils de base issus du projet GNU (GPL)
 - des bibliothèques et outils additionnels réalisant des fonctionnalités spécifiques (licences diverses, majoritairement GPL)
- L'appellation GNU/Linux traduit l'association du socle logiciel GNU au noyau Linux



GNU/Linux est un système d'exploitation

- Libre
- Complet
- Performant
- Versatile

En particulier, nous pouvons le configurer
spécifiquement pour des applications de sécurité

OpenBSD est un système d'exploitation
spécifiquement conçu et développé pour la
sécurité



La sécurité dans le monde GNU/Linux peut être envisagée sous deux axes :

- L'utilisation de GNU/Linux pour rendre des services (poste de travail, serveur, etc.) de manière sécurisée
- L'utilisation de GNU/Linux à des fins spécifiques de sécurité (Firewall, IDS, etc.)



GNU/Linux en tant qu'outil destiné à rendre des services...

- Socle système sécurisé (i.e. noyau)
- Compilations spécifiques et bibliothèques de sécurité
- Services sécurisé
 - **Design sécurisé**
 - **Programmation sécurisée**
 - **Fonctionnalité de sécurité intégrée**



Socle système sécurisé (et sécurisable)

- Noyau sain
- Fonctionnalité de sécurité intégrées au noyau (contrôle d'accès, environnements restreints, capabilities, etc.)
- Modèles de sécurité additionnels sous forme de modules (MAC, niveaux de sécurité, etc.)



Bibliothèques de sécurité

- Compilation d'outils permettant l'intégration de points de vérification au code existant
 - **Exploitation des failles plus difficile**
- Utilisation de bibliothèques de sécurité interceptant les appels jugés dangereux
 - **Limitation des comportements dangereux**



Logiciels “fiables”

- Design spécifique permettant de limiter les failles de sécurité dès la conception
- Contraintes de développement fortes pour assurer la sécurité du code produit
- Intégration de fonctionnalités de sécurité (contrôle d'accès) permettant de contrôler l'utilisation du service



GNU/Linux en tant qu'outil de sécurité

- Hérite des contraintes de sécurité précédente (un outil de sécurité est un service de sécurité)
- Doit disposer d'outils spécifiques selon le rôle attendu

→ Cf. 3e partie après la pause...

