

Le Logiciel Libre pour la Sécurité des Systèmes d'Information

Logiciels libres pour le déploiement d'une politique de sécurité
des systèmes et des réseaux

Cédric BLANCHER

cedric.blancher@eads.net -- <http://sid.rstack.org/>

Centre Commun de Recherche

Département SSI

Suresnes, FRANCE

JIA 2005, Sfax

9 fév. 2005



Introduction

Un logiciel libre est défini par 4 libertés fondamentales :

- la liberté d'utilisation (L0)
- la liberté d'accéder au code source et de l'étudier (L1)
- la liberté de redistribution (L2)
- la liberté de modification et de redistribution des modifications (L3)

Introduction

La sécurité d'un système d'information résulte d'un équilibre éclairé entre :

- le risque associé au SI qu'on protège (C1)
- le coût des protections mises en œuvre (C2)
- la pertinence des solutions déployées (C3)

Introduction

Cette présentation vise à montrer :

- Comment le logiciel libre répond aux besoins de sécurité
- Pourquoi le logiciel libre est pertinent en sécurité

- 1 La problématique de confiance
 - Origine du besoin de sécurité
 - La confiance dans le système d'information
 - Vers le logiciel de confiance ?
- 2 La problématique d'indépendance
 - La dépendance limite-t-elle la sécurité ?
 - La dépendance nuit-elle à la sécurité ?
- 3 Discussion ouverte
 - Validité des argumentaires
 - Axes de réflexion
- 4 Conclusion

- 1 La problématique de confiance
 - Origine du besoin de sécurité
 - La confiance dans le système d'information
 - Vers le logiciel de confiance ?
- 2 La problématique d'indépendance
 - La dépendance limite-t-elle la sécurité ?
 - La dépendance nuit-elle à la sécurité ?
- 3 Discussion ouverte
 - Validité des argumentaires
 - Axes de réflexion
- 4 Conclusion

Pourquoi la sécurité

Le besoin de sécurité découle d'un manque de *confiance*

- Je connecte mon SI à des entités inconnues
- Je ne peux pas contrôler ces entités
- Je ne fais pas confiance à ces entités

Dans un monde formidable type Disneyland (cf. débuts d'Internet), la sécurité n'était pas un besoin fondamental

Rétablir la confiance

La sécurité a pour but de *rétablir* la confiance

- La confiance en son système
- La confiance dans les utilisateurs
- La confiance dans les échanges avec l'extérieur
- Etc.

Vers la maîtrise

Paradoxalement, rétablir cette confiance revient à ne faire confiance à personne !

En fait, la sécurité rétablit la confiance dans le système à travers la *maîtrise* de l'environnement.

- 1 La problématique de confiance
 - Origine du besoin de sécurité
 - La confiance dans le système d'information
 - Vers le logiciel de confiance ?
- 2 La problématique d'indépendance
 - La dépendance limite-t-elle la sécurité ?
 - La dépendance nuit-elle à la sécurité ?
- 3 Discussion ouverte
 - Validité des argumentaires
 - Axes de réflexion
- 4 Conclusion

Les composants du SI

Un système d'information met des éléments en jeu

- Matériel
- Équipements "actifs"
- Systèmes d'exploitation
- Logiciels mis en œuvre
- Outils de sécurité (sens large)

La base de la confiance

La confiance se construit autour de brique de base

- Des acteurs (administrateurs, utilisateurs, etc.)
- Les composants du SI

La confiance dans le SI en général suppose la confiance en ces briques de base

Quelle confiance ?

Nous avons deux façons d'avoir de la confiance

- Décider arbitrairement qu'on peut faire confiance
- Se donner les moyens de vérifier qu'on peut faire confiance

Faire de la sécurité, c'est diminuer la confiance arbitraire (subjective) et augmenter la confiance vérifiée (objective).

- 1 La problématique de confiance
 - Origine du besoin de sécurité
 - La confiance dans le système d'information
 - Vers le logiciel de confiance ?
- 2 La problématique d'indépendance
 - La dépendance limite-t-elle la sécurité ?
 - La dépendance nuit-elle à la sécurité ?
- 3 Discussion ouverte
 - Validité des argumentaires
 - Axes de réflexion
- 4 Conclusion

La confiance dans le logiciel

On doit pouvoir faire confiance au logiciel. Comment ?

Règles de base de l'assurance qualité :

- Je dis ce que je fais (spécifications)
- Je fais ce que je dis (respect des spécifications)
- Je prouve ce que je dis (prouvabilité)

Aujourd'hui, *aucun* logiciel du marché ne répond à ces attentes fondamentales !

Pertinence

L'outil est-il adapté au besoin (C3) ?

- Vous savez ce que vous utilisez
- Ce que vous utilisez répond aux spécifications

Suppose une description factuelle et exacte du produit

Logiciel libre ?

La liberté L1 nous assure la disponibilité des sources

- Nous pouvons vérifier ce que fait le programme
- Nous pouvons vérifier comment il le fait

La possibilité de lire les sources est importante, la compilation essentielle !

Logiciel propriétaire

Nous ne disposons pas d'un accès libre au code source

- Impossibilité de vérifier ce que fait le programme
- Impossibilité de vérifier comment il le fait

Si on nous fournit le code source, l'accès est limité

- Impossibilité de régénérer le programme
- Impossibilité de valider la pertinence des sources

Confiance ?

Le logiciel libre nous permet de vérifier sa qualité.
Cette vérification diminue la nécessité de confiance subjective tout
en créant de la confiance objective
Elle augmente notre maîtrise du système (CQFD)

L'indépendance

Nous allons montrer qu'en sécurité :

- L'indépendance est nécessaire (mais pas suffisante)
- La dépendance nuit à la sécurité

- 1 La problématique de confiance
 - Origine du besoin de sécurité
 - La confiance dans le système d'information
 - Vers le logiciel de confiance ?
- 2 La problématique d'indépendance
 - La dépendance limite-t-elle la sécurité ?
 - La dépendance nuit-elle à la sécurité ?
- 3 Discussion ouverte
 - Validité des argumentaires
 - Axes de réflexion
- 4 Conclusion

Dépendance

La dépendance impose des interlocuteurs incontournables

- Éditeurs des logiciels choisis
- Revendeurs agréés par les éditeurs
- Fournisseurs de services agréés par les éditeurs
- Mainteneurs agréés par les éditeurs

Changer d'interlocuteur est difficile (retrouver une société agréés) et cher (si on change d'éditeur) et peut nuire à la pertinence des solutions

Indépendance

Indépendance vis-à-vis de l'éditeur :

- Patches de sécurité (L1+L3)
- Ajout de fonctionnalités (L1+L3)
- Suivi du logiciel (L1+L3)

Ceci permet d'assurer la pérennité de la solution (C2) de manière indépendante, en particulier en cas de fin de vie de produit (NT4.0) ou de disparition de l'éditeur...

Indépendance

Indépendance vis-à-vis du fournisseur :

- Disponibilité (L2)
- Maintenance (L1+L3)
- Déploiement (L0)
- Services (L1+L2+L3)

Là encore, on assure indépendamment la pérennité de la solution (C2)

Indépendance

Pas de licence contraignante :

- Conditions d'utilisation libres (L0)
- Pas de limite d'utilisation (L0)
- Pas de contrainte d'obtention (L2)
- Possibilité d'adapter le logiciel (L1+L3)

On assure l'universalité de la solution (C2+C3)

- 1 La problématique de confiance
 - Origine du besoin de sécurité
 - La confiance dans le système d'information
 - Vers le logiciel de confiance ?
- 2 La problématique d'indépendance
 - La dépendance limite-t-elle la sécurité ?
 - La dépendance nuit-elle à la sécurité ?
- 3 Discussion ouverte
 - Validité des argumentaires
 - Axes de réflexion
- 4 Conclusion

Dangers de la dépendance

Ce qu'il éviter

- Dépendance fonctionnelle (C3)
- Dépendance technologique (C3)
- Dépendance économique (C2)

Dangers de la dépendance

Dépendance fonctionnelle

- Maintenance
- Bug tracking
- Mises à jour (de sécurité)
- Nouvelles versions
- Intégration de fonctionnalités
- Etc.

La dépendance fonctionnelle nuit à la gestion efficace du SI, et le met en danger

Dangers de la dépendance

Dépendance technologique

- Incapacité d'évaluer la technologie
- Incapacité de qualifier sa sécurité
- Incapacité de valider une migration

On entre dans un cercle vicieux

Danger de la dépendance

Dépendance économique

- Problématique de l'Intelligence Économique
- Problématique des intérêts divergents
- Incapacité de maîtriser sa sécurité en terme de coût

- 1 La problématique de confiance
 - Origine du besoin de sécurité
 - La confiance dans le système d'information
 - Vers le logiciel de confiance ?
- 2 La problématique d'indépendance
 - La dépendance limite-t-elle la sécurité ?
 - La dépendance nuit-elle à la sécurité ?
- 3 **Discussion ouverte**
 - **Validité des argumentaires**
 - Axes de réflexion
- 4 Conclusion

Nombre de failles de sécurité

Argument difficile à appréhender et trompeur

- Système d'exploitation
- Outils de base du système d'exploitation
- Applications additionnelles tierces
- Etc.

Un système Windows comprend peu d'applications, et aucune tierces, alors qu'une distribution GNU/Linux en contient plusieurs milliers.

Présence de backdoor ?

Argument difficilement vérifiable, mais qui revient régulièrement

- Vraies backdoor (Lotus Notes "bridé" pour le gouvernement suédois)
- Spywares divers et variés
- Systèmes de prise de contrôle à distance pour la maintenance
- Mots de passe génériques (BIOS, HDD, Applications)

Un logiciel libre peut avoir une backdoor. Mais on peut la trouver et la supprimer...

Qualité de la sécurité

Mesurer la qualité de la sécurité, ne pas compter sur la "sécurité par l'obscurité"

- Cryptographie d'amateur (e-Book)
- Cryptographie non publiée (GSM, CSS, SDMI, etc.)
- Failles de sécurité découvertes et exploitées sans code source

La non publication des sources comme argument de sécurité est falacieux. On sait exploiter des failles IOS par exemple...

Informations sur les failles

Quel est notre niveau d'information sur les failles ?

- Le marketing l'emporte souvent sur le technique
- Capacité de juger de la criticité de la faille ?
- Capacité de trouver un contournement ?

Une information rapide et complète sur les failles est nécessaire

Correction des failles

Distribution des correctifs de sécurité

- Délai de publication (faille et correctif)
- Quid des failles considérées comme "non-critiques" (cf. Internet Explorer) ?
- Quid des produits en fin de vie (cf. Windows NT 4.0) ?
- Quid de l'impact des correctifs sur les systèmes de production ?

La résolution des failles est un énorme problème, épineux, qui ne devrait pas dépendre du bon vouloir d'un tiers...

Problèmes économiques

Le coût de la gestion du système (TCO)

- La dépendance impose souvent des coûts élevés
- Les systèmes de licence engendrent des coûts indirects
- La bonne maintenance suppose des systèmes destinés à la validation des patches (et des licences supplémentaires)
- Les formations constructeur/éditeur sont chères
- La documentation constructeur/éditeur est chère

Quelque soit le bout par lequel on prend la chose, la logiciel libre revient moins cher que le logiciel propriétaire sur la durée (cf. condamnation de Microsoft en GB pour publicité mensongère).

- 1 La problématique de confiance
 - Origine du besoin de sécurité
 - La confiance dans le système d'information
 - Vers le logiciel de confiance ?
- 2 La problématique d'indépendance
 - La dépendance limite-t-elle la sécurité ?
 - La dépendance nuit-elle à la sécurité ?
- 3 Discussion ouverte
 - Validité des argumentaires
 - Axes de réflexion
- 4 Conclusion

Les questions

Se poser les bonnes questions

- Quel est le niveau de confiance subjective acceptable ?
- Quel est le niveau de dépendance acceptable ?
- Quel est le degré de pertinence des outils proposés ?
- Quelle est la qualité du support disponible ?
- Quelle est la pérennité de la solution proposée ?

Autant de questions importantes auxquelles le logiciel libre peut répondre...

Le logiciel libre

Les réponses du Logiciel Libre

- On peut parvenir à un niveau de confiance objective maximal
- On peut parvenir à un niveau de dépendance nul
- On peut adapter les outils à ses besoins
- On peut faire réaliser le support par qui on veut
- On peut faire évoluer sa solution dans le temps

Windows NT 4.0 n'est plus maintenu depuis pas mal de temps, Windows 2000 est en fin de vie (juin 2005). Parallèlement à celà, les noyaux Linux 2.0.x sont encore maintenus...

Conclusion

La logiciel libre n'est pas une réponse universelle.
Ce n'est pas forcément la meilleure réponse.
Par contre, le logiciel libre offre des conditions qui permettent de satisfaire de manière pertinente de nombreux besoins spécifiques.

Questions ?

Ressources

Retrouvez cette présentation sur :

- <http://sid.rstack.org/>