Logiciels libres et formats ouverts pour la securité informatique.

Paris, Decembre 2000

Roberto Di Cosmo
Professeur
Université de Paris VII
e-mail: dicosmo@pps.jussieu.fr
WWW:
http://www.pps.jussieu.fr/~dicosmo

Free, Open Source Software : Logiciel libre

Libre=avec sources[2], Gratuit=pas payant

non libre, gratuit : Internet Explorer, MacTCP, Acrobat Reader, freeware etc.

non libre, non gratuit : tout le shareware, et logiciel commercial

libre, gratuit : Mozilla, Linux, FreeBSD, OpenBSD, sendmail, perl etc.

libre, non gratuit distrib. comm. Linux etc.

Quel espace économique pour le logiciel libre?

[2] *et* protegé par une *licence* libre autorisant modification et redistribution du code (ex: GPL, LGPL, MPL, BSD etc. à différents niveaux)

Logiciel libre: qui et comment gagne avec ce modèle

l'utilisateur actif (sait modifier le code)

- plus grande stabilité / pérennité / flexibilité
- transfert du budget support vers budget developpement
- partage des couts de developpement/maintien
- securite (on y reviendra)

Logiciel libre: qui et comment gagne avec ce modèle

l'utilisateur passif (ne modifie pas le code)

+ logiciel pratiquement gratuit
+ *Perennité* de la solution
+ pas d'emprisonnement proprietaire.
- cout de sortie des fois eleve (un hasard?)

Logiciel libre: qui et comment gagne avec ce modèle

les fournisseurs de services :

- support des utilisateurs actifs et passifs
- creation d'un espace economique de *proximité*.
- la richesse se crée dans le support, et reste acquise à qui la produit.
- *marge* et *marché* plus importants pour la SSII
- concurrence basée sur la *competence*

Logiciel libre: qui et comment gagne avec ce modèle

le dévéloppeur originaire :

non spécifiques :

- augmentation de la valeur du developpeur
- de ses biens
- création d'une demande de support,
- acceptation d'un standard (TCP/IP)

spécifiques :

- mutualisations couts off business core (gcc,systèmes embarqués etc.)
- valorisation de l'individu (propre à l'informatique)

Le coeur de la croissance est *la liberté*

Les protocoles *libres* (publics et ouverts):

- TCP/IP
- HTTP
- SMTP
- NNTP . . .

Tous les utilisateurs sont égaux. . .
face à la *transmission* des données. . .
partout sur la planète

Grande qualité de la communication grâce au libre accès aux *specifications* de protocoles non brevetés

Quels protocoles ouverts utilisez vous?

HTTP

```
C:GET /~dicosmo/index.html
S:  <html>
S:  <head>
S:  <title>Roberto Di Cosmo</title>
S:  </head>
S:  <body>
S:  <h1>Roberto Di Cosmo</h1>
S:  Professeur
S:  Ph.D. PISA, 1993<p>
S:  </body>
S:  </html>
```

Le coeur de la croissance est *la liberté*

Les formats *libres* (publics et ouverts):

- ISO-Latin
- *HTML*
- JPEG
- TeX/LaTeX/DVI . . .

Tous les utilisateurs sont égaux. . . face à l'*accés* aux données. . . *partout sur la planète*

Grande qualité du Web grâce au libre accès aux *sources* des documents
dont le format est non breveté `File/View Source...`

Le coeur de la croissance est *la liberté*

Les logiciels *libres*:

- sendmail
- bind
- Apache
- GNU software
- Linux/FreeBsd/NetBSD...

Tous les utilisateurs sont égaux...
face à l'accés à la technologie...
partout sur la planète

Grande qualité grâce au libre accès aux *sources non obfusquées* de logiciels
libres de brevet

Un logiciel libre plebiscité

Evolution des serveurs Web (Netcraft overall):

Hotmail.com a été réalisé avec à Apache, même aprés le rachat de Microsoft, qui a mis *3 ans* pour reussir à faire fonctionner Windows NT/IIS assez bien pour pouvoir se ”liberer” de ce logiciel libre.

Le coeur de la croissance vient de l'Europe

Quelques exemples:

bases : projet LeLisp à l'INRIA (France)

- $\Rightarrow$ Ilog
- $\Rightarrow$ NeXTStep (parti aux US)

WWW : 1989, CERN (Suisse), T. Berners-Lee, R. Cailleau sur NeXTStep

WebCrawler : 1992, US, avec l'IndexingKit de NeXTSTEP

Linux : 1990, Linus Torvalds, Finlande

Critères de choix pour l'entreprise

Coût

- coût de deployement
  - logiciel propriétaire: licences chères et syndrome du tapis roulant
  - logiliel libre: pas de licenses, dépense dans le service uniquement
- coût de maintenance:
  - logiciel propriétaire: très cher (tapis roulant), et hors du contrôle du client
  - logiciel libre: contrôle des source = le client est maître
- coût de sortie
  - souvent élévé pour toute reconversion, y compris 98 vers NT

Critères de choix pour l'entreprise

Enjeux stratégiques (hors coûts)

Seul le logiciel libre permet une maîtrise acceptable d'enjeux non directement monetisables

- sécurité des sytèmes (Word Spy/GUID/BackDoors/IE Spy etc.)
- perénnité de l'investissement (pas de phase-out pour cause du fournisseur)
- préservation de la propriété intellectuelle

LL: Intérêt pour la Sécurité

- Code source très largement disponible, utilisé et modifié par une masse d'ingénieurs système, donc
- étudié par des experts
  (track record: beaucoup plus de "defenseurs" que d"'attaquants")
- adaptable à des applications spécifiques, même en modèle de sécurité fermé
- prix trés competitifs (on ne paye que le service)

Mais le sujet est grave, et merite quelques reflections. . .

Modèles de sécurité

Fermé On construit un système dont les spécifications sont secrètes.
La sécurité repose en partie sur le secret des spécifications
(Ex: Enigma).

Avantages:

- On ajoute une difficulté supplémentaire pour les casseurs

Desavantages:

- La difficulté supplémentaire donne une fausse assurance
- Modèle traditionnellement sensible à la traison
- Difficile, voir impossible à mettre en oeuvre avec des composantes disponibles sur le marché

Modèles de sécurité

Difficile, voir impossible à mettre en oeuvre avec des composantes disponibles sur le marché,. . .

pourquoi?

- on ne maîtrise pas le code source propriétaire
- on ne sait pas faire d'audit de sécurité complet automatique sur 50M lignes de code
- chacune de ces lignes de code est suspecte
- même les outils de compilation sont suspects!

Modèles de sécurité

Ouvert les spécifications du système sont connues de tous.

La sécurité repose en partie sur la vérification par les paires (Ex: RSA).

Avantages:

- Modèle étanche à la traison (inutile de voler un decodeur RSA).
- Modèle facile á mettre en place avec des composantes du marché, à condition qu'elles soient aussi "ouvertes" (i.e. livrées avec leur code)

Desavantages:

- La qualité de la protection depende seulement de la qualité des algorithmes et de leur mise en oeuvre (crypto encore une art).
- La qualité de la protection depende aussi de la rapidité de correction des erreurs de conception ou mise en oeuvre (necessite encore du logiciel libre)

Proprietaire, Libre, Monopoliste

Mais le logiciel d'un monopoliste est encore moins adapté que celui d'une SSII privée quelconque à la securité: en effet,

- il multiplie les pirates (tout le monde a ce système)
- il divise les vérificateurs/correcteurs (presque personne a le code source).

Mais il y a plus que ça. . .

Conséquences de la mise en réseau

Toute information *peut* être

- *recueillie*
- *tracée*
- *transmise*

à notre insu

.

Microsoft Security Track Record

Microsoft (parmis d'autres) l'a fait, le fait et continuera à le faire pour plusieurs raisons:

lutte aux copies illégales :

- Windows Registration Wizard (`http://www.lemonde.fr/nvtechno/gates/pirate.html`)
- fichiers Office avec GUID
- projet Tempest

Proprietary Security Track Record

construction de profil commercial (int. opt.) :

- Cookies
- Champs HTTP-Referer
- Mobile Phones
- canaux cachés entre serveur et client Web
- . . .

Microsoft Security Track Record

négligence, stratégie anticompetitive :

- fichiers Word avec la trace de vos effacements
- *ActiveX*/Quicken/Web (Hambourg CCC),
  Java est (assez plus) sûr
- DLLs,
- backdoors
- virus, . . .

Microsoft Security Track Record

pressions US :

- Backdoors NSA dans les API crypto

N.B.: *tout logiciel proprietaire* est soumis à ces pressions...

... donc: les logiciels/protocoles/formats libres sont *indispensables* pour la transparence, et la *sécurité*.

Un exemple des dangers: Word espion

Document Word visible:

Voilà pour voir si on efface ou pas le contenu d'un document .... Si on autorise la sauvegarde rapide

Un peu gros?

```
ranger> ls -l Prova.doc
-rw-rw-rw-  1 dicosmo    22016 Mar  7 00:28 Prova.doc
```

Voyons...

```
ranger> tr -s '\000' ' ' < Prova.doc | tr -d '[:cntrl:]' > clair
```

Word espion

```
ranger> cat clair

Oh+’0p,8DPX‘hVoil pour voir si on efface Mi
oil Voyager oya Normal.dot v Voyager 4ya MicrosoftWord8.0 o@ˆ@z_g@
/gy-D3D9-11D2-85FC-D05649C10000}%A@
.+,D.+,Hhp|VMSVoilpourvoirsi
oneffaceTtulo6>_PID_GUIDAN{2D3B7143-D3D9-11D2-85FC-D05649C10000}%
A@Voila pour voir si on efface ou pas le
contenu d’un document....Si on autorise la sU\mH@AUXZ\:<>
@AUauvegarde rapide du document QUESTO NON VOGLIO CHE
SI VEDA 11111Qsdf qsdfqsdfQsdfqsdfqsdfsdfU\|mH@AUXZ\:<>
:<>@Z\|auve
garde rapide du document QUESTO NON VOGLIO CHE SI VEDA 11111Qsdf
qsdfqsdfQsdfqsdfqsdfsdfauv egarderapidedudocumentQsdfqsdfqsdfQsdfqs
```

Regardons mieux...

```
dfqsdfsdf[$@$NormalmHFA@FFuentedeprraf
opredeter.Aqrst
||U;Aqt?AprstVoyager#C:\Prova.doc @rr‘drrpTrtu
*^* gzLRootEntryT&Qdata.qsdA"2|T&Qd
Fgg’$D1TableQIA*"2c$$kD carte.QIF
CARTE.QIA&l.QIFPEL.QIA&’L.QIWordDocument eta.QIF
LIVRETA.QIF YPARAi$1i$IdecdA"1ndowSummaryIn
etc.
```

Hmmm…

Sécurité

La sécurité, on ne peut pas y penser *après*:

- Dos/W95/98 pas sûr par conception (monoutilisateur/(mono)tache)
- WinNT, en thèorie, bien conçu, mais ...
- Unix pas totalement sûr (origines BSD), mais contient les notions de base *depuis sa conception*
- Linux/xBSD les ont reprises, *et vous permettent de les élaborer*
- la biodiversité est *essentielle* (sauf pour votre dir. fin.)

Conclusions

- Les logiciels libres fournissent le seul profil acceptable pour les applications critiques.
  - immunes aux pressions externes
  - permettent l'inspection de leur code source, donc...
  - permettent la formation de vrais experts, donc...
  - multiplient les vérificateurs et non seulement les pirates
  - permettent une deploiement/adaptation à très faible coût
  - immunes aux décés par logique commerciales
  - ...
  - mais ils ne font pas (encore) de pubs à 20h à la télé