Principes et pratiques de sécurité FreeBSD

ccTLD Atelier

Décembre 2005
Dakar, Sénégal

Hervey Allen
traduit par Phil Regnauld
Network Startup Resource Center

FreeBSD vs. Linux

- Les modèles de sécurité sont en principe identiques, mais les implémentations sont différentes.
- Nous couvrirons des étapes spécifiques à FreeBSD.

Concepts de sécurité de base

Au final, le but est de:

- Préserver la confidentialité
- Protéger les données des intrus

- **Integrité -** protéger contre les pertes ou les modifications.
- **Authentification -**
  - Cette personne est-elle bien celle qu'elle prétend être ?
  - Cette personne a-t-elle le droit d'accéder à cette resources ?
- **Disponibilité –** Nos systèmes fonctionnent-ils ?

Préserver la confidentialité

Pour y parvenir:

- Utilisation correcte des droits utilisateurs et permissions fichiers
- Mot de passes robustes.
- Confiance dans les utilisateurs.
- Utilisation de bonne méthodes de chiffrement.

Protéger les données des intrus

Un certain effort est requis:

- Maintenir à l'extérieur les indésirables
  - Confiance dans les utilisateurs
  - Mots de passe robustes
  - Limiter les services qui ne devraient pas être actifs
  - Protéger/renforcer ceux qui le sont
- Chiffrer les données là où c'est nécessaire
- Sauvegarde en cas d'intrusion ou de corruption
- La sécurité physique est importante.

Integrité

Protéger vos données contre les pertes ou les modifications

- *Sauvegarde des données*
- Penser à la gestion des versions
- Détection d'intrusion (IDS)

Au bout du compte: les données ont-elles été modifiées par quelqu'un d'autre ? Comment le vérifier ?

Authentification

Comment s'assurer que...

- ...la personne qui utilise votre système est bien celle qu'elle prétend être ?
  - Confiance dans les utilisateurs
  - Mots de passes robustes
  - Systèmes à clé publique/privée
- ...la personne est autorisée
  - Bonne gestion des comptes
  - Permissions utilisateur/groupe/fichier correctes
  - Vérification des droits SUID et SGID

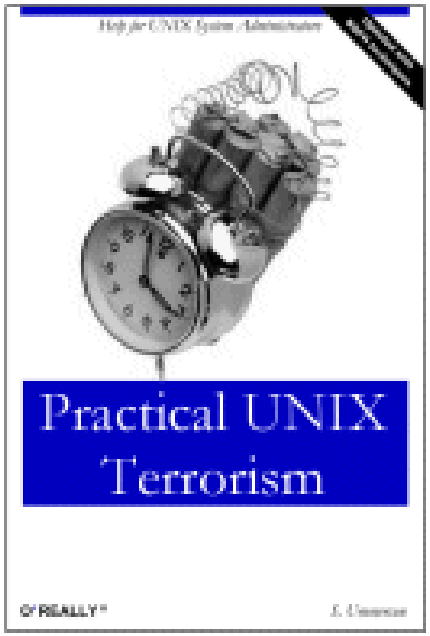
Disponibilité

S'assurer que les serveurs et services sont disponibles, détecter les dénis de service.

- Journalisation des services et analyse de ces journaux avec des “veilleurs” de log
- Configurer des alertes en cas de problème
- Détecter les attaques réseau: falsification (ARP, IP), inondations SYN, attaques force brute (attaques des services de login par dictionnaire: ssh, ftp, HTTP, ...).

Etapes

- Ne faire tourner que les services envisagés et nécessaires.
- Rester à jour et installer les correctifs nécessaires.
- Utiliser des mots de passes robustes et les enforcer.
- Contempler l'adoption de quotas.
- Restreindre les services d'accès à root
- Restreindre l'accès aux services via tcpwrappers si nécessaire



Etapes - suite

- Restreindre l'accès au serveur via les filtres IP (ipfw, ipf)
- Débordements de piles – être conscient des risques
- Journaliser les évènements, analyser
- Utiliser un système de détection intrusion
- Faites des sauvegarde
- Pensez à la sécurité physique
- Testez votre modèle de sécurité
- Ne pas oublier les machines clientes!

Quelques ressources:

Le manuel FreeBSD

- http://www.freebsd.org/doc/fr_FR.ISO8859-1/books/handbook/
- Chapitre 14: Security

Manuel de référence:

- *Practical UNIX and Internet Security*
- http://www.oreilly.com/catalog/puis3/

Une bonne source de documents avec exemples:

- http://nsrc.org/security/

Réduire le nombre de services

Qu'est-ce qui est démarré au boot ?

- `grep YES /etc/defaults/rc.donf`
- `grep YES /etc/rc.conf`
- `ls /usr/local/etc/rc.d`
- `/etc/inetd.conf`

Supprimer les services inutilisés

- Changer “YES” en “NO” dans `/etc/rc.conf`
- Supprimer les scripts de démarrage dans `/usr/local/etc/rc.d`
- Commenter les services inutiiles dans `/etc/inetd.conf` si inetd tourne.

Réduire le nombre de services actifs

Pour voir ce qui est actif:

- `lsof -i (si installé)`
- `netstat -an -f inet`
- `ps -auxw | more`
- `sockstat -4`
- `fstat (with grep, read man page)`

Comprendre chacun des processus

Supprimer tout service dont vous ne vous servez pas

Utiliser du chiffrement pour accéder à vos services

- POP/IMAP avec SSL uniquement.
- Envisager SMTP-TLS.
- Telnet est INTERDIT! SSH!
- Utiliser SCP ou SFTP et non FTP.
- Le FTP anonyme est OK, mais faire attention à ce qui est envoyé en upload
- Enforcer HTTPS pour les données sensibles.

Comment enforcer de bons mots de passe

Par défaut FreeBSD autorise des mots de passes faibles. Tester ceci avec `passwd` en tant qu'utilisateur..

Il exite un module PAM cracklib.

Cracklib empêche un utilisateur de choisir un mot de passe simpliste.

Où trouver cracklib:

- /usr/ports/security/cracklib

Pour l'ativer

- /etc/pam.d/passwd

Il suffit d'installer cracklib et d'activer une ligne dans /etc/pam.d/passwd.

Cracklib

Sortie de "`locate cracklib`" sous FreeBSD 5.4 après installation :

```
/usr/local/libdata/cracklib
  /usr/local/libdata/cracklib/pw_dict.hwm
  /usr/local/libdata/cracklib/pw_dict.pwd
  /usr/local/libdata/cracklib/pw_dict.pwi
  /usr/local/man/man3/cracklib.3.gz
  /var/db/pkg/cracklib-2.7_2
  /var/db/pkg/cracklib-2.7_2/+COMMENT
  /var/db/pkg/cracklib-2.7_2/+CONTENTS
  /var/db/pkg/cracklib-2.7_2/+DESC
  /var/db/pkg/cracklib-2.7_2/+MTREE_DIRS
```

Comme on le voit, un dictionnaire est installé.

Installation via"`pkg_add -r cracklib`" ou bien en compilant */usr/ports/security/cracklib*

Cracklib - suite

Extrait du README de cracklib

4) it's MIND-NUMBINGLY THOROUGH!

(is this beginning to read like a B-movie flyer, or what?)

CrackLib makes literally hundreds of tests to determine whether you've chosen a bad password.

It tries to generate words from your username and gecos entry to tries to match them against what you've chosen.

It checks for simplistic patterns.

It then tries to reverse-engineer your password into a dictionary word, and searches for it in your dictionary. *(> million entries!)*

- after all that, it's PROBABLY a safe(-ish) password. 8-)

D'autres vérificateurs de mot de passe

Des outils à lancer sur /etc/master.passwd régulièrement:

- John the Ripper: http://www.openwall.com/john/
- Crack: http://www.crypticide.org/users/alecm
- Le module PAM FreeBSD intégré pam_passwdqc.

Faire tourner un processus automaique qui vérifie régulièrement les mots de passes des utilisateurs. Les mots de passes “crackée” génèrent une alerte par mail indiquant à l'utilisateur de bien vouloir choisir un mot de passe plus robuste, ou se voir désactiver leur compte.

D'autres méthodes de contrôles des utilisateurs

- Regarder dans /etc/login.conf si vous souhaitez définir des classes d'utilisateur, et contrôler leur accès aux ressources.
  - FreeBSD Handbook section 13.7

http://www.freebsd.org/doc/fr_FR.ISO8859-1/books/handbook/users-limiting.html

- Envisager l'utilisation de quotas disque
  - FreeBSD Handbook section 16.14

http://www.freebsd.org/doc/fr_FR.ISO8859-1/books/handbook/quotas.html

Sauvegardez les données!

On ne peut le répeter assez souvent – si le système est compromis, comment faire sans backup ? Qui ici fait un backup régulier ?

Quelques éléments à prendre en compte:

- Que faut-il sauvegarder ?
- Quelle fréquence de sauvegarde ?
- Où se situent les médias de sauvegarde en cas de catastrophe (incendie, inondation, tremblement de terre, vol)?
- Que se passe-t-il en cas de perte totale ?
- Quel outils utiliser ? Tar, Arkeia, cpio, dump, dd, rsync et ssh?

Outils de sauvegarde

- **Arkeia**: produit commercial
  - http://www.arkeia.com/
  - http://nsrc/security/#backups
- **Bacula:** logiciel libre – recommandé!
  - http://www.bacula.org/
- **dd**: copie image d'un disque
  - `man dd`
  - `dd if=/dev/ad0 of=/dev/ad1 bs=32k conv=notrunc`

Copie image d'un disque vers un autre – ou vers un fichier:

```
dd if=/dev/ad0 of=/mnt/backup/ad0.img
```

Toujours faire attention à la source et à la destination!

Outils de sauvegarde

- **cpio**: créer des archives
  - cpitool: http://www.nickb.org/utils/
  - `man cpio`
- **dump**: sauvegarde du système de fichiers
  - `man dump`
- **rsync:** copie incrémentale à distance
  - `man rsync` (pas installé par défautt)
- **tar**: lire
  - `man tar`

Quelques exemples utiles

- Utilisation de tar + ssh pour faire une copie récursive vers une archive sur une autre machine:
  - `root@machine1# tar xzvf - /home/ | \ ssh machine2 “cat > machine1-homes.tgz”`
- Ou alors utilisation de rsync pour faire faire une copie d'une hiérarchie complète vers une autre machine:
  - `rsync -av . remote:/home/docs/`
- Utilisation de rsync –link-dest très intéressante.
- Voir aussi:
- http://www.cis.upenn.edu/~bcpierce/unison/

rsync avec ssh + clés ssh

Nous verrons plus tard l'utilisation de ssh avec des clés pour se passer de mot de passe.

Dans un script sous/etc/periodic/daily/ on effectue:

- `rsync -a /var/www/html/ \`
  `backup.machine:/var/www/html/`

Ceci copie de manière récursive le contenu de /var/www/html vers une machine distante – noter l'absence de -v

Si vous utilisez --delete alors les fichiers ayant été supprimés du côté de la source le seront aussi au moment de la mise à jour côté destination.

Evènements dans les journaux et analyse

Ceci prend du temps, même avec les outils disponibles.

Vous devez pour chacun des services décider si vous avez besoin de journaliser les évènements de ce service. Ceci est en partie établi dans /etc/syslog.conf sous FreeBSD.

Les journaux doivent si possible être envoyés vers une autre machine: un pirate essaiera d'effacer ses traces:

- syslog distant (“security.* @machine)
- chflags sappend fichier + securelevel

Evènements dans les journaux et analyse

- Penser à la comptabilité des processus:
  - man accton, acct, lastcomm, sa
- Toutes les commandes éxécutées sont journalisées.

Journalisation et/ou surveillance réseau

Quelques outils réseau utiles

- **Snort:** système de détection d'intrusion réseau, avec de bonnes règles de base. Tout sur Snort à http://www.snort.org/
- **Nagios**: surveillance service et serveurs – capable d'envoyer des mails ou des messages SMS en cas d'alerte http://www.nagios.org/.
- **nmap**: permet d'identifier les services ouverts au réseau sur une machine - http://www.insecure.org/nmap/.
- **ntop**: http://www.ntop.org/ présentation détaillées des statistiques réseaux, y compris résumé des interactions entre machines.

Attention: l'utilisation de certains de ces outils sur des réseaux autres que les vôtre peuvent vous apporter des ennuis!

Mise à jour des logiciels

- Au besoin appliquer des correctifs pour les logiciels / services que vous utilisez. Tenez-vous à jour via les listes de diffusion.
- Le vendeur de votre OS publiera régulièrement des correctifs.
- FreeBSD met régulièrement à jours les paquetages construits à partir des ports.

Quelques listes de diffusion

Sécurité générale

- BugTraq: http://www.securityfocus.com/
- CERT: http://www.cert.org/
- Rootshell: http://www.rootshell.com/

Pour Apache, Bind, Exim and SSH:

- http://www.apache.org/
- http://www.isc.org/ *(Bind)*
- http://www.exim.org/
- http://www.openssh.org/

Liste de diffusion des bulletins de sécurité FreeBSD:

- http://lists.freebsd.org/mailman/listinfo/freebsd-security-notifications

Penser à la sécurité physique

Toute la séecurité au monde ne fera rien contre les employés aigris, les serveurs exposés, les gens qui font des copies de clés... etc...

Sauvegardes: où sont-elles conservées ? Qui y a accès ? Sont-elles localisées au même endroit que le serveur ?

Journaux: sont-ils envoyés vers une machine distante ? Voire imprimés sur une imprimante ?

Chiffrement des donnés et disques: que se passe-t-il si quelqu'un part avec une machine sous le bras?! Ou juste le disque dur ? Sous FreeBSD: GEOM/GELI.

Accès physique = accès total

Envisager de resteindre l'accès à certains services

- Contrôle d'accès tcpwrappers via /etc/hosts.allow
- /etc/inetd.conf contrôle les services qui tourneront sous tcpwrapper.
- Activer /etc/inetd dans /etc/rc.conf avec:
  - inetd_enable=”YES”
- Que fournit ined ?

Que fournit inetd ?

- Le démon inetd écoute sur les ports de tous les services définis dans /etc/inetd.conf
- inetd économise de la mémoire en ne lançant les services que quand ils sont sollicités (peu efficace pour httpd!)
- Contrôle fin de comment les services sont lancés, et avec quelle fréquence, en utilisant les paramètres d'inetd.

inetd vs. ipfw/ipf

- Note: FreeBSD n'utilise pas xinetd.
- ipfw/ipf permettent un contrôle total des paquets entrants et sortants du système, y compris icmp et udp (tcpwrappers ne fonctionne que pour tcp).
- ipfw/ipf font partie du noyau, et sont donc très efficaces.
- la syntaxe d'inetd est plus simple à comprendre
- inetd sait envoyer un message pour les tentatives de connexion non-autorisées.

Inetd - suite

Si besoin est, on peut spécifier service-par-service à la fois dans/etc/inetd.conf et/etc/hosts.allow

Voir aussi:

- man inetd
- man hosts_access
- man hosts_options

Journalisation automatique

- Lire la page de man pour syslog – man syslog.conf
- FreeBSD envoie un rapport journalier lancé depuis cron (/etc/periodic), envoyé à root par défaut
- Envisager l'utilisation d'un serveur de logs centralisé -- voir syslog.conf.

Et encore plus de journalisation

Des outils utiles:

- **Swatch**: Simple WATCHer disponible sur http://swatch.sourceforge.net/ ou dans les ports / usr/ports/security/swatch. Attend l'apparition d'évènements clés dans les journaux et vous informe immédiatement
- **syslog et periodic:** voir "`man syslog`" et "`man periodic`" pour comprende comment les rapports d'activité réguliers sont générés dans FreeBSD.
- Voir http://nsrc.org/security/#logging pour quelques exemples d'outils

Envisager l'utilisation de quotas disque

Manuel FreeBSD sections 16.14 et Chap. 8:

- /usr/share/doc/en/books/handbook/quotas.html
- /usr/share/doc/en/books/handbook/kernelconfig.html

- Faites vous confiance à vos utilisateurs ?
- Que se passe-t-il si /tmp ou /usr (/usr/home) sont remplis ?
- Le FTP anonyme est-il actif ?
- Sont-ils sur des disques distincts ?
- Sinon, utilisez des quotas!

Exemples pratiques:

Exemples pratiques: quotas

De manière générale, pour activer:

- Recompiler le noyau avec "`options QUOTA`" dans le fichier de configuration noyau.
- Activer les quotas dans /etc/rc.conf avec:
  `enable_quotas="YES"`
- Activer les quotas utilisateur/groupe dans /etc/fstab:
  `/dev/da1s2g  /home  ufs rw,userquota,groupquota 1 2`
- Utiliser edquota pour configurer les fichiers quota.user et quota.group dans la racine de chaque système de fichiers sur lequel les quotas sont activés.
- Voir les commandes quota, quotaon/quotaoff, quotacheck, edquota, ainsi que /usr/ports/sysutils/setquota

Restreindre l'accès à root à un minimum de services

Chercher les fichiers avec bits suid/sgid. Enlever le bit (chmod u-s, chmod g-s) si il n'est pas utile ou suspect. Le rapport de sécurité FreeBSD lance cette recherche toutes les nuits.

Penser à restreindre l'environnement de certains services en utilisant chroot.

Lancer un services sous un utilisateur différent si possible.

Quelques conseils pratiques...

Conseils pratique de restriction de root

Pour trouver tous les fichiers suid:

- `find / -perm +6000 -type f -exec ls -ld {} \; > setuid.txt &`

Pruid une liste de tout les fichiers suid du système.

Pour désactiver les bits d'un fichier entièrement

`chmod 0nnn filename - suid`

`chmod 0000 filename – tous les bits`

Attention – FreeBSD est livré avec un nombre de fichiers suid/sgid par défaut, et vous alertera si de nouveaux fichiers apparaissent.

Conseils pratique de restriction de root - suite

- Utiliser `chroot` pour lancer des services dans leur propre “racine” – c'est à dire un “bac à sable” ou “prison”.
- FreeBSD dispose d'une fonction “`jail`”.
- Certains services fonctionnent déjà de manière restreinte: ntalk, finger, named, ...
- named est paramétrable dans /etc/defaults/rc.conf.
- Voir le manuel FreeBSD section 14.3.2 pour plus de détails.

Comment apache tourne-t-il sous l'utilisateur www ?

Extrait de */usr/local/etc/apache/httpd.conf*:

```
# If you wish httpd to run as a different user or group, you must run
# httpd as root initially and it will switch.
#
# User/Group: The name (or #number) of the user/group to run httpd as.
#  . On SCO (ODT 3) use "User nouser" and "Group nogroup".
#  . On HPUX you may not be able to use shared memory as nobody, and the
#    suggested workaround is to create a user www and use that user.
#  NOTE that some kernels refuse to setgid(Group) or semctl(IPC_SET)
#  when the value of (unsigned)Group is above 60000;
#  don't use Group "#-1" on these systems!
#
User www
Group www
```

Attaques par débordement de pile

Un pirate tente d'envoyer dans le tampon mémoire d'un programme plus de données qu'il y a d'espace réservé. Ils peuvent corrompre des structures de données en mémoire dans l'espace du programme et faire éxécuter des commandes avec les mêmes privilèges que ceux du service compromis.

Un grand nombre de correctifs adressent ces problèmes particulièrement.

Le monde UNIX/Linux a un nombre de solutions à ce problème, qui est très répandu, avec certaines pénalités (cf. OpenBSD)

Configuration et utilisation d'un IDS

- Intrusion Detection System = IDS
- Network Intrusion Detection System = NIDS
- Equivalent: System Integrity Checking (contrôle d'intégrité système)

Un IDS contrôle le trafic réseau et vous alerte si du trafic suspect est détecté.

Un contrôle d'intégrité système vérifie les modifications inattendues de fichiers, et vous alerte de ce fait (par exemple: AIDE, http://www.cs.tut.fi/~rammer/aide.html)

Pour une liste d'outils: http://nsrc.org/#integrity

Snort SDI

Snort (http://www.snort.org/) est un SDI très populaire, et sert à détecter les activités réseaux irrégulières, en utilisant un jeu de règles et de signatures (séquences). C'est ce qu'on appelle un SDI à base de signature.

Parmi les modules additionnels pour SNORT (certains sont maintenant intégrés dans SNORT):

- **ACID**: Anaylsis Console for Intrusion Databases. Interface web pour la gestion des alertes et rapports. Bon pour un gros site. Voir http://acidlab.sourceforge.net/.
- **Sguil**: Système avancé permettant d'analyser des évènements de SDI avec des outils tels que ethereal et TcpFlow et bien sûr Snort. Voir http://sguil.sourceforge.net/.
- **Snort_inline**: http://snort-inline.sf.net/. Détecter les tentatives d'intrusion et y réagir.
- **SnortSam**: http://www.snortsam.net/mise à jour en temps réel des coupe feus en réponse à des tentatives d'intrusion.

Restreindre l'accès à votre serveur à l'aide du service de filtrage (ipfw)

FreeBSD 5.4 est équipé de pas moins de 3 solutions de filtrage IP:

1.) IPFIREWALL: ou *ipfw*. version 2, (ipfw2), standard depuis 5.0. Exemple un peu dépassé dans le manuel, voir plutôt / etc/rc.firewall – mais lire man ipfw

2.) IPFILTER: ou *ipf* - multiplateforme, assez bien documenté dans le manuel

3.) Packet Filter: ou *pf* importé d'OpenBSD. Très avancé, inspiré de IPFILTER -- recommandé.

Présentation détaillée:

- /usr/share/doc/en/books/handbook/firewalls.html

Filtrage IP (suite)

Extrait du manuel (section non traduite)

La configuration du logiciel IPFW se fait par le biais de la commande IPFW(8). La syntaxe de cette commande a l'air compliquée, Mais elle est relativement simple dès que l'on comprend la logique.

Il y a actuellement 4 catégories de commande: **addition/suppression**, **affichage**, **vidange**, and **remise à zéro**. L'addition/suppression sert à construire des règles qui contrôlent comment les paquets sont acceptés, rejetés, et journalisés. L'affichage sert à visualiser le contenu de vos règles de filtrage, et les compteurs pour chaque règle (paquets et octets – comptabilité). La vidange sert à supprimer toutes les règles d'un jeu. La remise à zéro sert à annuler les compteurs des différentes règles (comptabilité).

Filtrage IP (suite)

Pour utiliser IPFW il faut configurer /etc/rc.conf et modifier /etc/rc.firewall

Il est recommandé de journaliser les paquets pour aider au débogage des règles.

Quelques exemples de règles:

```
ipfw add deny tcp from les.mechants to les.gentils 22
ipfw add deny log tcp from 10.20.30.0/24 to les.gentils
```

Explication sur la page suivante...

Filtrage IP (suite)

Cette règle empêche tout paquet provenant de la machine les.mechants d'atteindre la machine les.gentils sur le port 22, en TCP (ssh)

```
ipfw add deny tcp from les.mechants to les.gentils 22
```

L'exemple suivant rejette et journalise tout trafic TCP depuis le réseau 10.20.30.0/24, à destination de la machine les.gentils (quel que soit le port):

```
ipfw add deny log tcp from 10.20.30.0/24 to les.gentils
```

Filtrage IP (suite)

Avant de commencer:

- Lire la section “Firewalls” du manuel FreeBSD
- Lire “`man ifpw`” - “`man ipf`” - “`man pf`”
- Voir le guide très instructif de PF ici:
  - http://www.openbsd.org/faq/pf/fr/index.html

- La conception d'un jeu de règles complet peut être complexe.
- Les instructions dans le manuel FreeBSD sont *excellentes* et il est fortement recommandé des les lire.

Testez votre modèle de sécurité

Tentez de vous connecter depuis l'extérieur et contrôlez que votre sécurité est correcte!

Lancez un programme de scan réseau contre votre machine.

Par exemple: `nmap`.

Un autre outil est nessus, que nous allons voir ci-après.

Voir:

- http://www.insecure.org/nmap/
- http://www.nessus.org/.

Testez votre sécurité: nmap

Attention! Ne pas lancer nmap contre un réseau distant avant de prévenir et/ou demander autorisation.

Essayons maintenant de scanner la machine du voisin:

- `nmap 10.20.30.NN`
- `nmap -O 10.20.30.NN`
- `nmap -sS -O -p 1-1024 -v \ 10.20.30.NN`

Lire la page de manuel de `nmap` pour comprendre de quoi il ressort. Pour une bonne introduction à `nmap` voir:

`http://linuxgazette.tolix.org/issue56/flechtner.html`

Ne pas oublier vos clients!

S'assurer que vos utilisateur soient obligés d'utiliser des méthodes sécurisées pour se connecter à vos serveurs.

Insistez sur l'utilisation de SSH plutôt que telnet, SCP/SFTP plutôt que FTP, POP/IMAP via SSL, etc...

Il faut prendre en compte les problèmes de sécurité que représente Windows tels que les virus, les mises à jour Windows Update, les spywares/chevaux de troie/vers.

Un scanner de virus pour contrôler le mail ? Filtrer également les extensions suspectes.

Problèmes sociologiques. La sécurité peut être une gêne. Par exemple, Windows n'intègre *toujours* pas de client ssh – c'est pénible.

Nous évoquerons Windows XP et 2000 et quelques conseils de sécurité pratique.

Quelques liens

CERT (Coordinated Emergency Response Team)

- http://www.cert.org/ and http://www.us-cert.gov/cas/index.html

Bonne liste de resources sécurité pour Linux/UNIX

- http://www.yolinux.com/TUTORIALS/LinuxSecurityTools.html

nmap: outil d'exploration réseau et scanner de sécurité

- http://www.insecure.org/nmap/

Livres O'Reilly

- http://www.oreilly.com/

SANS Sécurité informatique et listes de diffusion

- http://www.sans.org/ et http://www.sans.org/newsletters/risk/

Documents sécurité sur nsrc.org

- http://nsrc.org/security/ et http://nsrc.org/freebsd-tips.html

Et ne pas oublier qu'il existe une commaunté qui peut vous aider à http://www.afnog.org/!

Autres resources

Le manuel FreeBSD, section sécurité:

- http://www.freebsd.org/doc/fr_FR.ISO8859-1/books/handbook/security.html

Ports FreeBSD “intrusion detection”

- http://www.freebsd.org/cgi/ports.cgi?query=intrusion+detection&stype=all

FreeBSD liste de diffusion des bulletins de sécurité

- http://lists.freebsd.org/mailman/listinfo/freebsd-security-notifications

Logiciel d'audit de sécurité Nessus

- http://nessus.org/

Conclusion

Plus de sécurité peut impliquer moins de convivialité, mais une brèche de sécurité est très peu conviviale.

Il y a un équilibre à atteindre entre la quantité de sécurité en place et les services que vous offrez.

Vos utilisateurs se plaigneront, mais ils râleront encore plus si leurs données sont compromises – n'oubliez pas de leur rappeler ceci!