CONSEILS DE SECURITE SUR L'ADMINISTRATION DE MACHINES UNIX SUR UN RESEAU TCP/IP

Version 4 (mars 92)

Jean-Luc Archimbaud CNRS/UREC
chargé de mission “sécurité informatique” au CNRS

La sécurité informatique n’est plus un domaine réservé à la Défense Nationale. Sans devenir paranoïaques, responsables et utilisateurs des systèmes d’information doivent s’appliquer à limiter les risques qu’encourent leurs données et leurs matériels.

Il faut considérer que s'il n'y a pas de responsable sécurité désigné :

Le Responsable de la sécurité d’un système est l’administrateur de ce système

Ce document est donc destiné aux administrateurs de stations de travail Unix, pour les aider dans cette tâche. Il a pour but de fournir les conseils de base pour rendre une machine Unix, connectée sur un réseau TCP/IP, moins vulnérable aux pirates. C’est une compilation de différentes publications, volontairement limitée à n’être qu’un “**livre de recettes**” au détriment de l’analyse des failles et des remèdes.

Il est limité à l’aspect logiciel (Unix), et n’aborde pas les conseils généraux de protection contre le feu, le vol, ... Il ne s’adresse ni aux gourous Unix, ni aux spécialistes de sécurité informatique.

Unix est un système d’exploitation créé par des programmeurs, pour des programmeurs, dans un laboratoire de recherche. La sécurité n’a donc pas été une préoccupation dominante lors de sa conception. Mais, plus que d’un atavisme, sa vulnérabilité provient principalement de :

• Sa popularité : c’est le système d’exploitation le plus connu. Beaucoup de pirates ont exploité et exploiteront les bugs d’un système dont le source est facilement disponible.

• L’attitude des vendeurs : ils livrent un système totalement ouvert.

1. CONSEILS AUX ADMINISTRATEURS

En tant que Responsable de la sécurité sur votre machine, vous devez effectuer les vérifications nécessaires et mettre en place les outils fournis par les constructeurs pour la protéger. Les paragraphes suivants listent ces outils. Mais la technique est loin de protéger intégralement votre système. L'attention et la vigilance sont vos 2 principales armes. Les militaires ont coutume de dire que "La sécurité, c'est 20% de technique et 80 % de bon sens".

Et, avant toute chose, n'oubliez pas cette conduite à tenir :

En cas de problème de piratage, avertissez immédiatement votre responsable hiérarchique

1.1. VERIFICATIONS LORS DE LA MISE EN SERVICE DE LA MACHINE

Après l'installation de votre système d'exploitation sur disque (à la livraison ou lors d'un changement de version), faîtes une sauvegarde complète et un "***ls -Rl***" que vous conserverez. Ceci pourra vous servir de référence.

1.1.1. /etc/passwd et /etc/group

Sur ces fichiers :

• Vérifiez que le propriétaire est root avec les accès 444; et que tous les utilisateurs et tous les groupes sont déclarés avec un **mot de passe** (le second champs de chaque entrée ne doit pas être vide).

• Si vous n'utilisez pas NIS (les "Yellow Pages"), supprimez la ligne dont le premier caractère est + (si elle existe).

• Si vous l'utilisez, vérifiez que les lignes " **+ : :0 : :0 : : :**" et "**+** :" sont absentes dans les fichiers de la station serveur. Elles ne doivent être présentes que sur les machines clientes.

Dans **/etc/passwd :**

• Seul l'administrateur doit posséder 0 comme "user id" (UID).

• Modifiez les mots de passe fournis avec le système ainsi que le mot de passe pour root que vous avez utilisés lors de l'installation.

• Otez les utilisateurs de service (guest, visitor, tutor, demo, ...). Vérifiez que les comptes anonymes (sys, uucp, bin, adm, lp, ...) ont "*" comme mot de passe.

• Utilisez les extensions des nouvelles versions d'Unix sur la gestion des mots de passe. Exemples :

Pour que les mots de passe chiffrés n'apparaissent pas dans le fichier passwd (shadow password)

Pour limiter la durée de validité (aging) des mots de passe. Choisissez une durée qui est un bon compromis en fonction du profil de vos utilisateurs.

1.1.2. Fichiers sensibles

• Sous **/dev** :

L'accès au directory doit être 755. Le super utilisateur "root" doit être le propriétaire de tous les fichiers, exceptés les fichiers relatifs aux terminaux actuellement login.

Les fichiers kmem, mem et les partitions des disques (avec les noms sd*, rxy*, ... selon le matériel) doivent avoir l'accès 0 pour "other".

• Vérifiez que le directory **/lost+found** possède l'accès 700

• Mettre l'accès t à tous sur **/tmp**: *chmod o+t /tmp*

• Si votre système offre la possibilité de "**secure terminal**", mettez la en pratique : enlevez le mot "secure" dans chaque ligne du fichier **/etc/ttytab** (ou /etc/ttys). L'absence de cet attribut proscrira le

login direct sous root. Le passage en super utilisateur se fera uniquement par la commande ***su***.. Spécifiez, aussi les quelques utilisateurs qui pourront faire su root lors de la déclaration du groupe wheel dans /etc/group.

• Supprimez "." (le directory courant) dans les règles de recherche de l'utilisateur root (la variable **path** est initialisée dans /.cshrc et PATH dans /.profile).

• Le fichier **/.rhosts** est très dangereux. Si vous ne l'avez pas créé en pleine connaissance de cause et s'il existe, détruisez le. Si vous l'utilisez, vérifiez ses accès et son contenu.

• Effacer le fichier **/etc/hosts.equiv** si vous n'en avez pas l'utilité. Dans le cas contraire, vérifiez que ce fichier ne contient pas une seule ligne d'un seul caractère : +.

• Si vous n'utilisez pas NFS, ou si vous ne désirez pas rendre accessible (exporter) une partie de votre arborescence, détruisez le fichier **/etc/exports.** Dans le cas contraire, vérifiez que chaque nom de directory que vous désirez "exporter" est suivi des noms des stations qui ont le droit d'y accéder. Autrement, n'importe qu'elle station pourra "monter" ce directory. La syntaxe dépend des versions d'Unix (faites : *man exports*).

• Enlevez, s'ils sont présents, les alias "decode" et "uudecode" dans le fichier **aliases** sous /etc ou /usr/lib.

• Supprimez l'accès w à "other " sur les fichiers aliases, aliases.dir et aliases.pag sous /etc ou /usr/lib.

• Dans le ficher **sendmail.cf** (sous /etc ou sous /usr/lib), vérifiez que la variable W (wizard password) a pour valeur le caractère étoile. Concrètement, dans ce fichier, la ligne commençant par OW (la lettre O suivie de la lettre W) doit être de la forme : OW*

• **cron** :

Limitez autant que faire ce peut l'accès au batch d'Unix (commandes crontab et at). Créez sous /usr/lib/cron (ou sous /usr/spool/cron) les fichiers cron.allow et at.allow avec uniquement une ligne contenant la chaîne de caractères "root" (uniquement root pourra utiliser cron). A la demande, vous ajouterez les utilisateurs qui ont besoin du cron dans ces fichiers.

Supprimez tous les accès à "other" sur le fichier spool/cron/crontabs/root sous /usr ou /usr/var. Vérifiez que root est le seul utilisateur à posséder l'accès w sur toutes les procédures lancées par ce fichier.

• Vérifiez que l'exécutable **chroot** (sous /usr/bin ou /usr/etc) possède l'accès 700

1.1.3. inetd.conf

Le daemon **inetd**, toujours actif, offre des services tels que telnet, rlogin, ftp, ... Ces services, qui s'exécutent sur votre machine, sont accessibles depuis les machines du réseau. Seuls les services déclarés dans le fichier **inetd.conf** (sous /etc ou /usr/lib) ou /etc/**servers** répondront aux machines distantes. N'hésitez pas à faire du ménage dans ce fichier, en ajoutant un # devant les services que vous n'utilisez pas.

Ce peut être :

• **ruserd** : permet à un pirate de connaître les utilisateurs connectés sur votre station.

• **tftpd** : version simplifiée de ftp, il est principalement utilisé pour charger le système d'équipements sans disque (terminal X par exemple) via le réseau. Indiquez un nom de directory après l'argument -s dans la ligne qui lance le daemon. Si l'argument -s n'est pas prévu, sachez que votre version de tftpd est vieille et possède un gros bug de sécurité.

De manière plus restrictive encore, si votre station n'est jamais accédée depuis une autre machine (donc dans le sens entrant), vous pouvez supprimer :

• **fingerd** : très utile au demeurant (pour obtenir le numéro de téléphone d'un utilisateur, par exemple); il peut permettre à un pirate, depuis une machine quelconque du réseau, d'avoir des informations sur les utilisateurs de votre station.

• **rlogind** : gère les rlogin entrants

• **rshd** : gère les exécutions de commandes sur votre machine, lancées depuis une machine distante
• **rexecd** : le pendant de rshd pour les fonctions
• **telnetd** : gère les accès interactifs à votre machine par telnet
• **ftpd** : gère les transferts de fichiers par ftp initialisés depuis une machine distante
• **rpc.* :** répond aux requêtes RPC (Yellow Pages, NFS, ...) venant de machines distantes
• **talkd** : gère l'échange de messages (commande talk)
•**rwalld** : accepte les messages générés par rwall depuis une station du réseau

Ces suppressions n'affectent pas le sens sortant. Vous pourrez toujours, depuis votre station, utiliser telnet, ftp, ...

1.1.4. /etc/rc*

Les scripts **/etc/rc*** lancent des daemons réseaux, similaires à inetd. Les scripts livrés avec les systèmes ont la fâcheuse tendance d'activer des daemons inutiles. Il est préférable de ne pas les lancer (en ajoutant un # devant les lignes add hoc du script) si vous n'en avez nul besoin. Entre autres :
• **rwhod** : diffuse régulièrement à toutes les machines du réseau des informations concernant les utilisateurs login sur votre station.
• **sendmail :** pour recevoir du courrier (mail) venant d'autres machines du réseau (l'option de debug distant d'anciennes versions de sendmail est dangereuse). Attention, ce daemon est obligatoire pour la messagerie inter-machines.
• **routed** : met à jour et diffuse des tables de routage IP de manière automatique et incontrôlable.
• **nfsd** : pour être serveur NFS.
• **biod** : pour être client NFS.

1.2. ACCES TCP/IP

1.2.1. Rappels sur TCP/IP

Sur un réseau TCP/IP, pour accéder à un service offert par une machine (interactif, transfert de fichiers, nfs, rwho, lpr ...) il faut que ce service soit ouvert : le daemon doit être lancé et vous devez posséder les autorisations nécessaires (mot de passe, ...). Mais, avant cette phase, il faut que les machines puissent dialoguer en TCP/IP. C'est ce que l'on peut appeler "l'**accès TCP/IP**". En limitant les possibilités d'accès TCP/IP à votre machine, vous minimisez d'autant les risques de piratage.

Avec TCP/IP, l'accès est toujours symétrique. Si vous pouvez accéder à une machine X, un utilisateur sur cette machine X pourra accéder à votre matériel. Inversement, si vous ne pouvez pas accéder à X, X ne pourra pas vous atteindre. Donc, si vous ne pouvez pas atteindre un réseau, vous êtes certain que les machines de ce réseau ne pourront pas vous pirater. Par contre, si vous pouvez accéder aux ordinateurs du monde entier ... L'accès TCP/IP est la conséquence du routage que vous installez sur vos stations.

1.2.2. Routage IP

Les routages IP installés sur votre machine (visualisés par la commande *netstat -r*) déterminent les accès TCP/IP de votre machine. Voici quelques conseils :
• Ne mettez en place que les routages nécessaires aux accès dont vous avez besoin.
• Sauf si vous connaissez RIP, supprimez le daemon **routed** lancé dans un des scripts /etc/rc*. Utilisez de préférence le routage manuel avec des commandes *route*.
• Mesurez bien la conséquence d'une route par défaut (commande ***route add default*** ...) : toutes les machines TCP/IP du Monde peuvent essayer d'entrer sur votre système.
• De manière plus drastique, vous pouvez utiliser le routage par machine. Ainsi la commande : *route add 129.89.32.2 ...* vous permettra de communiquer avec cette machine particulière, sans ouvrir l'accès à toutes les machines du réseau 129.89.

1.2.3. Passerelle avec l'extérieur

Si votre laboratoire dispose d'un réseau Ethernet TCP/IP, raccordé sur un réseau fédérateur (d'un campus ou d'une région) lui-même connecté sur un réseau national, ... il peut être sage d'installer un équipement avec 2 coupleurs Ethernet entre votre réseau interne et le réseau fédérateur. Il sera **routeur IP**, **passerelle** entre votre laboratoire et le monde extérieur. Ce peut être un matériel dédié ou une banale station de travail

En limitant le routage IP (en ôtant, par exemple, le daemon routed sur la passerelle et en ajoutant aucune commande *route add* ...), les stations internes ne seront jamais inquiétées. Il vous suffira de surveiller l'accès à cette passerelle; et de n'y installer aucun daemon ou utilitaire dangereux et aucune donnée confidentielle. Mais attention, un utilisateur qui a pu entrer sur la passerelle pourra accéder aux machines internes.

1.3. OPERATIONS A EFFECTUER REGULIEREMENT

1.3.1. Sauvegardes

De manière évidente, une bonne politique de sauvegardes périodiques est impérative pour la sécurité de votre système. Il faut pouvoir revenir à un état antérieur propre et sûr. Ne négligez pas la protection des supports de sauvegardes contre les risques physiques (effacement, vol, feu ...).

Utilisez de préférence ***dump*** et ***restore*** qui sont réservés à l'administrateur.

Pensez que vous pouvez autoriser un groupe d'utilisateurs (des opérateurs par exemple) à effectuer des sauvegardes, sans leur donner le mot de passe de root. Il suffit d'écrire un tout petit programme qui lance les sauvegardes, dont le propriétaire sera root, avec le **setuid** positionné (chmod u+s) et exécutable par le groupe add-hoc.

1.3.2. Mot de passe

Dans **/etc/passwd** :

• Otez les utilisateurs qui ne travaillent plus sur votre machine. Détruisez aussi tous les fichiers de ces utilisateurs. La commande suivante liste les fichiers qui appartiennent à personne (en fait les fichiers dont l'UID du propriétaire n'est plus dans /etc/passwd).

find / -nouser -o -nogroup -print

• Vérifiez que tous possèdent un mot de passe et que 2 utilisateurs n'ont pas le même UID.

• Vérifiez que le caractère "+" n'a pas disparu de la ligne "**+ : :0 : :0 : : :**" si elle existe.

Rappelez à vos utilisateurs (par mail ou par motd) de **changer leur mot de passe**.

1.3.3. Root

Parcourez l'historique des login (et des su) de root. Ces traces sont stockées dans des fichiers tels que **messages*** ou **sulog** ou **loginlog** sous /usr/adm ou /var/adm. Vous pouvez ajouter l'affichage du contenu de ces fichiers dans votre .login (ou l'équivalent).

1.3.4. Vérifications sur certains fichiers

Vérifiez que :

• Les fichiers .profile, .cshrc, .login ... sous la racine ne sont pas accessibles en écriture à tous.

• Il n'y a pas de fichier étrange dont le nom commence par un "." sous /tmp et /usr/tmp.

• Le contenu de /etc/**hosts.equiv** et de /etc/**exports** est correct.

• Les exécutables des programmes **su**, **login** et **telnet** sont ceux d'origine.

• Les fichiers lancés par **cron** pour root ne présentent aucune anomalie (dans /usr/var/spool/cron/crontabs/root ou ...)

• Il n'y a pas pléthore de fichiers avec l'accès w à "other" dans l'ensemble de votre système, avec la commande :

find / -type f -perm -2 -exec ls -al {} \;

1.3.5. Sushi

Un **Sushi** (Super User SHell Interactive), permet à un utilisateur d'être sous le shell avec tous les privilèges de root. C'est le programme shell, appartenant à root, avec le Set-User-ID bit (SUID) positionné. Pour dépister un Sushi, vérifiez régulièrement que les fichiers qui appartiennent à root avec le Set-User-Id bit sont uniquement des utilitaires. Il ne doit pas y avoir ce genre de fichier sous une arborescence d'utilisateur. La commande :

find / -user root -perm -4000 -exec ls -al {} \;

permet de lister ce type de fichier.

1.3.6. .rhosts et .netrc

Contrôlez les accès (700 de préférence) et le contenu des fichiers .rhosts chez vos utilisateurs. Ils permettent d'entrer sur votre système sans mot de passe local. Pour afficher le contenu de ces fichiers, vous pouvez utiliser la commande : ***find /users -name .rhosts -print -exec cat {} \;***

Vérifiez que aucun utilisateur a créé un fichier .netrc sous son home directory.

1.4. CONSEILS GENERAUX

1.4.1. Habitudes de travail

• Le **mot de passe de root** est la clef qui ouvre toutes les portes : choisissez le après réflexion, changez le très régulièrement, ne le divulguez pas.

• Faites *logout* chaque fois que vous quittez votre poste de travail.

• Réservez le **login sous root** à l'administration du système. Utilisez un autre login lorsque vous n'avez pas besoin de privilège.

• Ne laissez jamais une autre personne travailler sous root, même pour quelques minutes.

• Faîtes ***/bin/su*** au lieu de *su* pour accéder à root.

• Dans votre fichier .login ou .profile, rajoutez la commande ***who***. Elle peut vous permettre de détecter des utilisateurs qui ne devraient pas être présents sur votre machine.

• Utilisez de préférence les utilitaires d'administration fournis avec votre machine, plutôt que l'accès direct aux fichiers de configuration avec un éditeur.

• Utilisez un compte spécial, avec le minimum de privilège, lors des démonstrations, essais, ... en présence d'autres personnes.

1.4.2. /etc/hosts.equiv

Avec hosts.equiv vous déléguez entièrement les contrôles de sécurité aux machines citées dans ce fichier. Vous faîtes alors totalement confiance à d'autres administrateurs. Ceci est très dangereux. Donc, sauf besoin très particulier, proscrivez l'utilisation du fichier hosts.equiv (détruisez le).

1.4.3. Groupes

• Si des utilisateurs désirent partager des fichiers, ne les laissez pas céder à la facilité de l'accès rw à "other" sur les fichiers. Les possibilités des groupes (cf /etc/group, /etc/passwd, *umask 007*, *chown, chmod, newgrp*, ...) résolvent ce problème d'accès partagé.

• Chaque compte déclaré sur votre machine doit correspondre à une et une seule personne physique clairement identifiée. Vous devez posséder les coordonnées de chaque utilisateur, avec son type d'activité sur votre machine.

1.4.4. Divers

• Installez régulièrement les nouvelles versions de votre système d'exploitation : elles corrigent les erreurs de sécurité et les **nouvelles versions d'Unix** sont de plus en plus sécurisés.

• Installez les nouvelles fonctions qui obligent à posséder un mot de passe pour pouvoir relancer une station en mode mono-utilisateur (**single user**), état où l'utilisateur possède tous les privilèges). Si votre système n'offre pas cette possibilité (mot de passe sur PROM sur SUN, clé sur IBM ...), il faut néanmoins prendre en compte ce risque, en contrôlant par exemple l'accès physique à la machine.

• Vérifiez toujours le contenu d'un **archive**, avant d'extraire les fichiers qu'il contient.

• Evitez d'utiliser **UUCP**, trop ancien avec trop de trous de sécurité.

• Ne créez un programme avec **SUID root** qu'avec beaucoup de précaution et que si vous êtes un programmeur expérimenté. Protéger le source du programme. Ne faites pas de shellscript SUID root.

• Si vous installez un **ftp anonymous** sur une de vos machines, ne le mettez pas sur votre serveur principal et créez un environnement restreint (cf *man ftpd*)

• Sensibilisez vos utilisateurs aux problèmes de sécurité. La première action est de diffuser largement le chapitre "Conseils aux utilisateurs".

• Les conseils aux utilisateurs ci-après doivent aussi être suivis par les administrateurs.

2. CONSEILS AUX UTILISATEURS

Une règle officielle et primordiale : si vous découvrez un piratage, un essai de piratage ou un état suspect :

avertissez immédiatement l'administrateur de la machine et votre responsable hiérarchique

2.1. Mot de passe

- Prenez du temps pour le choisir.
- Changer le régulièrement.
- Ne reprenez pas un mot de passe que vous avez déjà utilisé.
- Modifiez le avant de partir en vacances.
- Ne l'écrivez pas, ne le confiez à personne.
- Ne choisissez pas :
    - Votre nom, prénom ou celui de vos proches.
    - Une information personnelle : numéro de téléphone, ...
    - Un mot contenu dans un dictionnaire.
    - Une variation (inversion, initiales, ...) sur les 3 types précédents.
- Un bon mot de passe doit être composé :
    - d'au moins 6 caractères.
    - d'un mélange de majuscules, minuscules, chiffres et caractères de ponctuation.
- Exemple : c'est1KO

Pour découvrir un mot de passe, le pirate ne teste pas toutes les combinaisons de caractères. Il s'appuie sur les habitudes de l'utilisateur moyen. Il essaie, entre autres, toutes les informations relatives à l'utilisateur (nom, ... avec les variations) et les mots du dictionnaire.

Si vous accédez à une machine à travers des réseaux, sachez que votre mot de passe circule en clair sur les réseaux que vous traversez.

2.2. umask

La commande umask permet de créer un masque qui définit les **modes d'accès attribués par défaut aux nouveaux fichiers créés** (cf man umask)**.** Généralement, ce masque est initialisé à 002 ou 022 pour tous les utilisateurs du système, ce qui correspond à l'accès 775 ou 755. Ainsi, par défaut, tout nouveau fichier est accessible en lecture à tous. Ceci est trop ouvert.

En ajoutant la commande *umask* dans votre .cshrc ou .profile, vous pourrez modifier l'accès par défaut. N'hésitez pas à utiliser *umask 077* qui assurera une meilleure protection sur les fichiers que vous créerez.

2.3. Travail en groupe

Si vous devez partager un même environnement de travail avec plusieurs collaborateurs, demandez à l'administrateur d'enregistrer tous les membres de votre équipe sous un **même groupe**. Pour stocker les **fichiers partageables** sont un répertoire "Commun" :

- Créez un sous-répertoire qui contiendra les objets partageables : *mkdir Commun*
- Donnez les accès rwx au groupe sur ce répertoire : *chmod g+rwx Commun*
- Pour éviter certains conflits d'accès : *chmod g+t Commun*
- Rajoutez dans votre .login ou .profile : *umask 007*

2.4. Divers

Faites logout chaque fois que vous quittez votre poste de travail (ne serait ce que 5 minutes).

Dans les régles de recherche (**path** ou PATH), spécifiez le répertoire courant "." en fin de liste et non au début.

Lorsque vous accédez à votre machine, lisez attentivement l'heure et la date de votre **dernière connexion**, information généralement contenue dans la bannière d'accueil.

Utilisez à bon escient .**rhosts** et vérifiez régulièrement son contenu.

Sachez que l'administrateur d'une machine (root) peut lire tous vos fichiers et votre boite à lettres.

3. DOCUMENTS

Documents ayant inspirés ce “livre de recette” :

- Improving the security of your unix system
  SRI-David A. Curry
- Security features guide
  Sun Microsystems OS 4.0.3
- The Internet Worm Program : An Analysis
  Eugene H. Spafford
- Unix System Administration (chapitre security)
  David Fiedler & Bruce H.Hunter
- Advisories
  CERT-CC
- Guide de sécurité pour les administrateurs de systèmes Unix
  Christian Pelissier
- Site Security Handbook (RFC1244)
  P. Holbrook & J. Reynolds