Sécurité sous UNIX

Constantin Yamkoudougou

28 décembre 2004

1 Abbréviations

BSD : Berkely Sofware Distribution
BIOS : basic Input / Output system
ATT : American Telephon and Telegraphs
DARPA : Defense Advanced Research Project Agency
IETF
MIT : (Massachusetts Institute of Technology) OSF : Open Software Consortium
IEEE : Institute of Electrical and Electronics Engineers
POSIX : Norme UNIX de l'IEEE, numérotée 1003.1, et qui spécifie le noyau du système.

La sécurité des systèmes UNIX est relativement complexe à aborder compte tenu de la diversité des UNIX dans ce domaine. Dennis Ritchie [ref 1], un des contributeurs importants de première génération et concepteur du langage C disait ceci à propos de la sécurité UNIX **" it was not designed from the start to be secure. It was designed with the necessary characteristics to make security serviceable"**. Un système UNIX par défaut n'est donc pas sécurisé. La diversité des systèmes UNIX se justifie par l'histoire même de ce système d'exploitation.

ref 1 http ://cm.bell-labs.com/cm/cs/who/dmr/

2 Bref rappel sur la genèse d'UNIX

A l'image d'Internet, la préhistoire d'UNIX commence dans les années 60 avec le programme de recherche de système d'exploitation baptisé MULTICS [ref1] par l'agence DARPA . Les caractéristiques essentielles de ce projet étaient la protection des utilisateurs les uns des autres et la définition de plusieurs niveaux de sécurité. La DARPA [ref 2] (Defense Advanced Research Project Agency), le MIT (Massachusetts Institute of Technology) et ATT (American Telephone and Telegraph) étaient les principaux acteurs du

projet MULTICS d'origine.

En 1969 ATT se retira du projet DARPA mais un des chercheurs Ken Thompson poursuivit des développements internes sur ce qui allait devenir le futur système UNIX. La philosophie générale des chercheurs d'ATT était de réaliser un système modulaire en réalisant des programmes unitaires très performants qui accompliraient une tâche à la fois. En effet, l'un des reproches que l'on pouvait faire à MULTICS était sa vocation à réaliser simultanément des taches multiples et complexes.

En octobre 1973, le système UNIX fut complètement réécrit par Thompson grâce au langage C créé par Richie. A partir de 1977, on comptait déjà plus de 500 sites utilisant un UNIX ATT dans le monde.

En acquérant les sources du système, L'unversité de Berkeley allait jouer un rôle majeur dans l'évolution d'UNIX à partir de 1978 grâce à des developpements importants de Bill Joy et Chuck Haley. Ces nouveaux développements allaient donner naissance à la version UNIX BSD [ref 3] (Berkeley Software Distribution).

Compte tenu de la grande popularité que connût la version BSD, le staff d'ATT crayant de perdre le contrôle sur UNIX proclama la nouvelle version d'alors (UNIX ATT système V) comme standard et qualifia l'UNIX BSD de non standard et d'incompatible. Ce schisme UNIXien allait donner naissance à deux familles UNIX. D'une part l'UNIX ATT system V et d'autre part les dérivés BSD adoptés par des éditeurs tels que DEC et Sun.

En 1988 ATT et Sun tentèrent de fédérer un système UNIX compatible avec les sources UNIX BSD et ATT System V pour donner naissance au système Solaris encore connu sous le nom de SunOS 5.5.

Suite à cet accord entre ATT et Sun, plusieurs autres équipementiers dont Hewlett - Packard, IBM, DEC créèrent le consortium OSF (Open Software consortium) pour tenter d'éviter le contrôle d'UNIX par un oligopole et d'en assurer ainsi la gestion par une association à but non lucratif. Les lourdeurs de fonctionnement de L'OSF n'ont pas permis la conception de standard, si bien que certains membres comme IBM créèrent leur UNIX connu sous le nom d'AIX.

Aujourd'hui on dénombre au moins une dizaine d'UNIX selon les éditeurs et les projets Open source. La gestion et l'evolution d'UNIX est désormais dévolue à l'IEEE par le bias du standard POSIX, connu aussi sous le nom de ISO / IEC 9945 [ref4].

ref 1 http ://www.multicians.org/f7y.html
ref 2 http ://www.darpa.mil/
ref 4 http ://www.UNIX.org/version3/iso_std.html
ref3 http ://www.computerhope.com/UNIX/bsd.htm

3 Et Linux ?

Linux est un système d'exploitation initité en 1991 par Linus Torvalds [ref1] qui s'interessait au système MINIX alors étudiant de l'université d'Helsinki . MINIX [ref 2] est un système UNIX réduit a des fins pédagogiques par le professeur Andrew S Tanenbaum[ref 3]. La version Linux 0.01 publiée par Linus Torvalds supportait un shell : le bash et le compilateur gcc. De part son histoire Linux est un système UNIX. Son noyau est developpé sous forme de logiciel libre avec plusieurs déclinaisons de packages logiciels cohérents appelés distributions [ref 4] .

ref 2 http ://groups.google.com/group/comp.os.minix/
ref 3 http ://www.cs.vu.nl/ ast/
ref 4 http ://www.linux.org/dist/list.html
ref 1http ://www.li.org/linuxhistory.php

4 Avantages d'un système UNIX dans le système d'information d'entreprise

Un système informatique est sécurisé si l'on peut en déterminer le comportement et le mode de fonctionnement selon une certaine politique de sécurité. L'intérêt d'un système UNIX par rapport à d'autres systèmes d'exploitation du marché, est sa granularité, sa stabilité et la complète transparence grâce à la mise à disposition de sources pour les UNIX libres. Si les coûts d'acquisition s'avèrent quasiment nuls pour les droits d'acquisition dans le monde libre, l'investissement d'une entreprise dans ce domaine peut en revanche être important en ressources humaines pour les tâches d'exploitation et de maintenance qui peuvent requérir des compétences spécifiques.

5 Précautions minimum en matière de sécurité système Unix

Compte tenu de la diversité des Systèmes UNIX, une étude au cas par cas est nécessaire pour chaque système. Parmi les principes de base de gestion de

la sécurité de ces systèmes, on peut citer la gestion des fichiers journaux, la sécurité du système de fichiers, de la pile TCP/IP, la gestion du démarrage des serveurs, la gestion des mots de passe...

Gestion de fichiers journaux
Sous UNIX deux types de gestion des fichiers journaux sont possibles. Une gestion locale et une gestion centralisée. Le processus demon syslog permet de gerer finement les fichiers journaux. Il est toujours recommandé de gérer de manière centralisée les journaux pour des raisons de sécurité et aussi pour en faciliter les audits.

Contrôle au démarrage
Pour les éléments sensibles du système d'information tels que les serveurs, un contrôle des redémarrages systèmes et du chargement des programmes est nécessaire. Ce contrôle peut s'exercer par l'activation d'un mot de passe BIOS requis à chaque amorçage du système. En effet un amorçage non controlé peut conduire à la prise en charge intégrale d'un système par un tiers.

Configuration minimale des systèmes
Il s'agit de doter le système UNIX des processus indispensables au système d'information. Tout les processus superflux non indispensables aux utilisateurs doivent en principe être supprimés ou rendus inactifs. Les techniques de prise de contrôle communément exploitées par les pirates consistent le plus souvent à rechercher des vulnérabilités sur des processus très peu connus qui sont ensuite exploitées (via l'exécution de shell code par exemple) pour prendre la main sur des programmes plus importants.

Sécurité du système de fichiers
Les droits sur un système de fichier UNIX sont organisés selon 3 triplets de trois bits (rwx) chacun. Ces trois triplets représentent respectivement les droits de trois types d'identité : le propriétaire du fichier, le groupe et les tiers. Le premier bit permet de positionner les droits de lecture, le second les droits d'écriture, le troisième les droits d'exécution. Pour le cas des fichiers exécutables, le bit d'exécution peut être forgé à la valeur particulière s, pour le propriétaire du fichier ou le groupe. On utilise le terme suid ou sgid pour désigner ces états de fichier. Le fichier peut alors être non seulement exécuté par son propriétaire mais aussi par le groupe du propriétaire. Une mauvaise gestion des droits s notamment pour les fichiers appartenant au compte administrateur peut conduire à la prise en main d'un système UNIX.

6 Contôle d'accès et de la sécurité des mots de passe sous UNIX

Bon nombre de systèmes UNIX permettent l'utilisation de modules PAM (Pluggable Authentication Module) [ref 1] dont le rôle ne restreint plus uniquement à l'authentification mais une gestion fine de chaque service vis à vis des utilisateurs ou groupe d'utilisateurs du service . PAM est supporté par la plupart des UNIX tels que AIX, Solaris, HP-UX, BDS et la plupart des distributions Linux.

En matière de gestion de mot de passe, la plupart des systèmes UNIX sont configurés par défaut pour utiliser le programme **passwd** qui est de plus en plus abandonné par les administrateurs au profit de **Npasswd** [ref 2] permettant de définir une granularité plus fine conforme aux politiques de mot passe pouvant être définies dans les entreprise [ref3] .. Au delà de ces utilitaires de gestion de mot de passe, la sécurité du contrôle d'accès UNIX par mot de passe dépend évidemment de la complexité des mot de passe utilisés par les utilisateurs. Un bon mot de passe consiste en une suite d'au moins 8 caractères. Composés d'au moins une lettre majuscule [A-Z], de lettres minuscules [a-z], d'au moins une caractère numérique[0-9], d'au moins un caractère spécial [@#$% &*;;?...]. L'usage de mots émanant du dictionnaire est à proscrire ansi que la répétition de login ou une partie du login dans le mot de passe. Sont aussi à proscrire l'usage de prenoms, noms de famille, dates d'anniversaire etc. Une bonne politique de sécurisation des accès par mot de passe consiste également en la définition d'un délais d'expiration du mot de passe. Il est par exemple possible de définir le renouvellement des mots de passe par période de 15, 30 ou 45 jours. Il est nécessaire d'associer à cela une règle qui permette si possible de ne pas rejouer le même mot de passe.

ref 1 http ://docs.linux.com/article.pl
ref 2 http ://www.utexas.edu/cc/UNIX/software/npasswd/
ref 3 réference à la fiche PICSI comment définir, mettre en place et auditer une politique de mot de passe.

7 Sécurité de la pile réseau sous UNIX

La sécurité de la pile réseau sur un système UNIX est à l'image du protocole TCP/IP qui par essence n'a pas été conçu pour assurer la confidentialité ou l'intégrité des paquets transmis sur le réseau. Par conséquent, les usurpations d'adresses source ou destination, de numéros de ports ou encore la corruption de cache sont autant d'actes malveillants possibles sur un système UNIX- Des utilitaires de sécurité ont été développés au fil du temps

pour pourvoir à l'absence de sécurité dans le protocole IP. Parmi ceux-ci, on peut citer les projets libres suivants :

ipf ou **IP Filter** [ref1] en plus de son rôle de firewall, permet aussi de faire de la translation d'adresse en IPv4. Il est compatible avec les UNIX de la famille BSD. Le projet **IP Filter** a été initié par Daren Reed.

ipfw [ref2] développé par l'université de Berkeley pour le système FreeBSD remplit les mêmes fonctions que **IP Filter**.

SOCKS standard IEFT [ref3] conçu pour être utilisé à la fois comme proxy applicatif et outils de firewall mais aussi comme outils de mise en oeuvre de réseaux privés virtuels.

netfilter/iptables , utilitaire de filtrage spécifique au systèmes Linux de noyau 2.4 et 2.6

ref1 http ://coombs.anu.edu.au/ avalon/
ref2 http ://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/firewalls-ipfw.html
ref3 http ://www.socks.nec.com/AboutSOCKS/WhySOCKS.asp
ref 4 http ://www.netfilter.org/

8 sécuriser les appels systèmes et le noyau

Le noyau est le coeur d'un système UNIX. Il assure l'intermédiation avec l'architecture matérielle (processeur, périphériques) et les programmes. Le noyau orgainise l'exécution des programmes au sein de processus. Il charge les bibliothèques dynamiques associées à chaque programme. Pour éviter que le noyau ne charge des librairies corrompues, il est possible d'utiliser des systèmes de détection d'intrusion. Par exemple SYSTRACE [ref1] est un outils permettant de faire un filtrage d'appel système bas niveau. La configuration d'un tel outils peut se faire à l'image d'un firewall pour lequel par défaut tout ce qui n'est pas explicitement autorisé est interdit.
Par ailleurs au delà de tout système de protection, la mise à jour des noyaux et l'application régulière de patches de sécurité est indispensable pour se protéger des nouvelles vulnérabilités [ref 2][ref 3].. La mise à jour du noyau permet aussi de bénéficier de nouvelles fonctions et pilotes (exemple compatibilité IPv6).

ref 1 http ://www.systrace.org/
ref 2 http ://www.securiteam.com/
ref 3 http ://www.securityfocus.com/

références documentaires complémentaires