Sujet 110 : Sécurité

- 110.1 Effectuer des tâches concernant la sécurité au niveau utilisateurs (Weight 3)
- 110.2 Configurer la sécurité du système (Weight 3)
- 110.3 Sécuriser des échanges réseau avec le cryptage (Weight 3)

Sécurité des utilisateurs

- Vérifier les droits **SUID** et **SGID** : Supprimer les fichiers dangereux

```
find / -user root -perm -4000 -print | /bin/mail -s "Setuid root files" root
```

-
- Vérifier les packages installés : **rpm -Va**
- Expiration des comptes et des mots de passe : **chage**
- Interdire la connexion : **/bin/false, /etc/nologin, /etc/shadow, passwd, usermod.**

- Les droits SUDO
  - Fichiers de configuration : **/etc/sudoers**
  - Edition **: visudo**

Sécurité des utilisateurs (2)

- **Les limites de l'utilisateur** : Agir sur l'environnement du shell et des processus qu'il contrôle :
  - Taille de fichier, nombre de fichiers ouverts, nombre de processus, la taille de mémoire max que l'utilisateur peut occuper etc ...
  - **Limite Soft** et **Limite Hard**.
  - **Limite Hard** est défini par le super-utilisateur pour un utilisateur ou un groupe d'utilisateurs et ne peut pas être dépassé.
  - **Limite Soft** est également défini par le super-utilisateur, mais il peut être remplacé temporairement par un utilisateur en cas de besoin avec la a commande **ulimit.**
  - **Limite Soft** et **Limite Hard**.sont définies **/etc/security/limits.conf**
  - Valeurs par défaut **: ulimit -a**
  -

Ports ouverts

- Liste des ports ouverts en local : **lsof, netstat**
- Outil d'audit des ports ouverts à distance : **nmap**

```
# nmap -A machine
Starting Nmap 4.20 ( http://insecure.org ) at 2008-05-22 19:54 CEST
Interesting ports on machine.mondomaine.com (192.168.1.25):
Not shown: 1676 closed ports
PORT     STATE SERVICE     VERSION
21/tcp   open ftp          vsftpd 2.0.5
22/tcp   open ssh          OpenSSH 4.3p2 Debian 9 (protocol 2.0)
25/tcp   open smtp         Postfix smtpd
...
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.14 - 2.6.17
....
```

Contrôler les services

- Trou de sécurité dans un service bien particulier,
- Mauvais paramètrage d'un service :
  - Votre MTA devient un «open relay » utilisé par les spammers, open relay blacklists
  - sshd acceptant les connexions root de l'Internet
- Désactiver les services que vous n'avez pas besoin :
  - Service standalone : **chkconfig** , **update-rc.d**
  - Service géré par le super démon **: inetd / xinetd**

Super démon inetd

- Fichier de configuration **/etc/inetd.conf**

```
# <service_name> <sock_type> <proto> <flags> <user> <server_path>
   <args>
# To re-read this file after changes, just do a 'killall -HUP inetd'
#

ftp     stream tcp   nowait root     /usr/sbin/tcpd in.ftpd -l -a
telnet  stream tcp   nowait root     /usr/sbin/tcpd in.telnetd
```

Super démon xinetd

- Intégrer les fonction du démon tcpd et rendre plus lisible et flexible la configuration.
- Fonctionnalités de gestion des log
- Fichier de configuration globale : /etc/xinetd.conf :
- Répertoire contenant les fichiers spécifiques aux services : /etc/xinetd.d/

```
defaults
{
        instances           = 60
        log_type            = SYSLOG authpriv
        log_on_success      = HOST PID
        log_on_failure      = HOST
        cps                 = 25 30
}
includedir /etc/xinetd.d
```

tcp_wrappers

- **tcp_wrappers** permet la vérification des accès à un service réseau.
- Le binaire du service est compilé avec la librairie statique **libwap**.
- La vérification des accès à un service réseau a lieu en trois étapes :
  - **l'accès estil explicitement autorisé ?**
  - **sinon, l'accès estil explicitement interdit ?**
  - **sinon, par défaut, l'accès est autorisé.**
- Fichiers de configuration : **/etc/hosts.allow** et **/etc/hosts.deny.**
- Syntaxe :
  - `daemon_list : client_list`

Secure Shell (SSH)

- Remplacer les services de connexion à distance vulnérables : **ftp, rlogin, rcp, et telnet.**
- **OpenSSH** est une version LIBRE du protocole SSH développé par **OpenBSD**
- Boite d'outils : **ssh**, **scp**, **sftp**, **ssh-keygen, ssh-add, ssh-agent** ...
- **Version 1**
  - Se limiter à une simple fonctionnalité de connectivité à distance pour le shell.
  - Une faille permettant à un pirate d'insérer des données dans le flux chiffré
- **Version 2**
  - Sécurisation de n'importe quel protocole applicatif et ceci grâce à ses mécanismes de « port forwarding » et de « tunneling ».
  - sftp
  - Normalisée en 2006 à l'**IETF**. **RFC 4250 à 4256**

Installation

- Programme serveur : **sshd, port 22**
- Programmes client : **ssh, scp, sftp,**
- Boite d'outils **ssh-keygen, ssh-agent**
- Fichier de configuration du serveur **sshd_config**
- Fichier de configuration du client **ssh_config**
- Couple de clés de la machine
  - **ssh_host_rsa_key** et **ssh_host_rsa_key.pub**
  - **ssh_host_dsa_key** et **ssh_host_dsa_key.pub**
  - **ssh_host_key** (ssh version 1)
- Known hosts : **~/.ssh/known_hosts et /etc/ssh/ssh_known_hosts**

GPG (Gnu Privacy Guard)

- **Clone libre de PGP (Pretty Good Privacy).**
- **ali**
- Générer le couple de clés : `ali@bagdad:~$ gpg --gen-key`
- Exporter la clé publique : `ali@bagdad:~$ gpg --export aloulou > gpg.pub`
- **Salah**
- Importer la clé publique : `salah@bagdad:~$ gpg --import gpg.pub`
- Gérer le trousseau des clés : `gpg --list-keys`
- Crypter :

```
gpg --out cryptogramme --recipient aloulou --armor --encrypt message
```

- **ali**
- Décrypter :

```
ali@bagdad:~$ gpg --out lisible --decrypt cryptogramme
```

-

GPG : Signature numérique

- **ali**
- Signer : `ali@bagdad:~$ gpg --clearsign lisible`
- **salah**
- Vérifier Signature : `salah@bagdad:~$ gpg --verify lisible.asc`
-