INSA – 5° R.T.

Sujet d'examen
20 janvier 2004

Sécurité des systèmes d'information

2ème partie

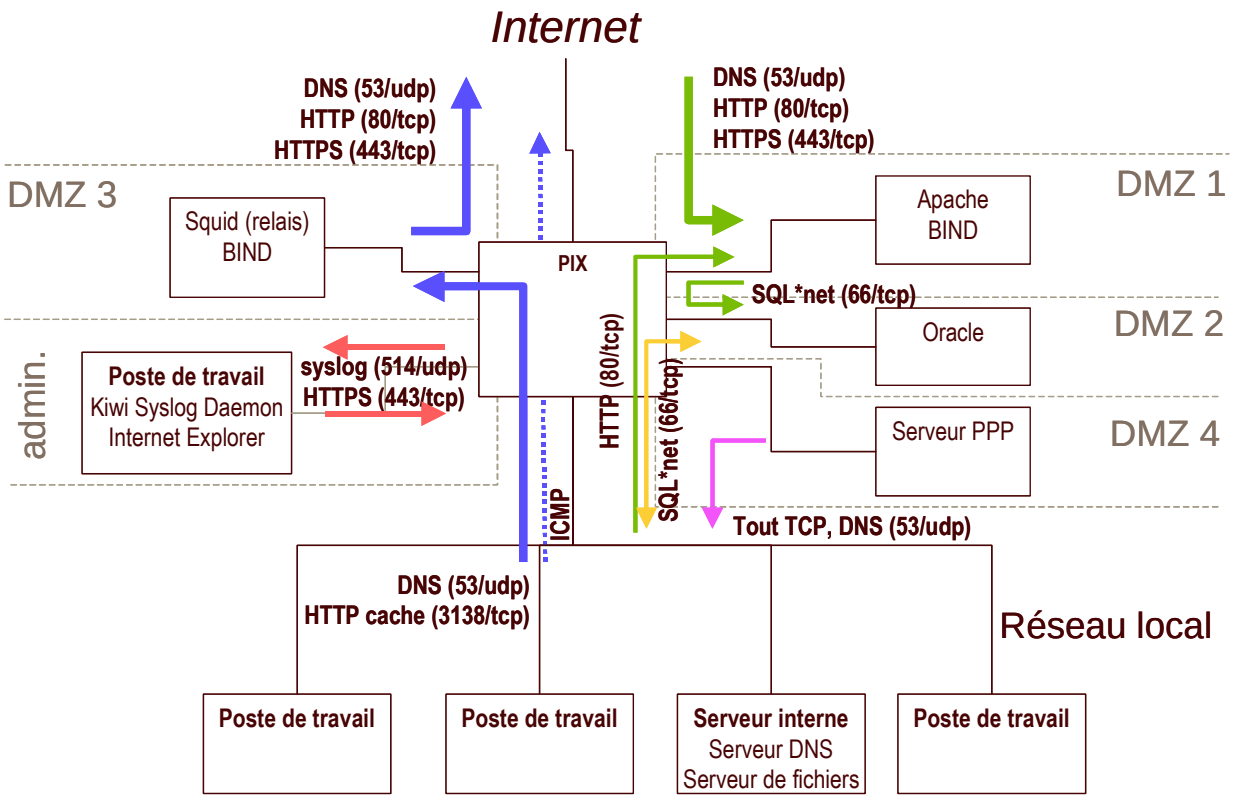
Exercice 1

Donnez un exemple d'action concrète pour chacune des grandes fonctions d'un RSSI telles qu'elles vous ont été présentées :

1. Définition de la politique de sécurité
2. Analyse de risques
3. Sensibilisation et formation aux enjeux de la sécurité
4. Etude des moyens et préconisations
5. Audit et contrôle
6. Veille technologique et prospective

1. *Définition de la politique de sécurité*
    a. *Interview avec les différents cadres responsables du S.I.*
    b. *Rédaction de la politique de sécurité*
    c. *Discussions avec les partenaires (service du personnel, représentants du personnel, direction) pour la mise en place de la traçabilité*
2. *Analyse de risques*
    a. *Identification de la valeur des informations*
    b. *Evaluation du coût d'un arrêt du système*
3. *Sensibilisation et formation aux enjeux de la sécurité*
    a. *Présentation d'une technique de choix d'un mot de passe*
    b. *Prise en main des postes de travail (mires d'authentification, accès distant, etc.)*
    c. *Clarification de la notion de signature d'une clef publique*
    d. *Apprendre aux utilisateurs à faire la différence entre un HOAX (faux message d'alerte) et un vrai message de l'antivirus de messagerie*
    e. *Informer sur la nécessité de ne pas diffuser les adresses email*
4. *Etude des moyens et préconisations*
    a. *Rédaction d'un guide de configuration système*
    b. *Préconisation de règles de collectes des traces*
    c. *Maquette de systèmes d'authentification biométriques*
    d. *Evaluation des performances de différents firewall*
5. *Audit et contrôle*
    a. *Utilisation d'un outil d'audit automatique (type Nessus)*
    b. *Recherche/Validation des services réseau ouverts*
    c. *Exploitation des traces collectées (recherche d'accès nocturnes, etc.)*
    d. *Attaque des mots de passe*
6. *Veille technologique et prospective*
    a. *Suivi des alertes CERT*
    b. *Suivi des vulnérabilités identifiées par les constructeurs*
    c. *Autres techniques d'authentification*

Exercice 2



Question 1 : Compte tenu du mode de fonctionnement suggéré par le schéma, indiquez l'usage de chacune des DMZ et la (les) fonction(s) qu'elle apporte à l'organisation utilisant cette architecture. Pour chaque fonction (ou à nouveau pour chaque DMZ si vous préférez), critiquez (avantages/inconvénients) la solution.

Question 2 : En ce qui concerne les flux entrants ou sortants du réseau local (depuis ou vers les DMZ ou Internet), indiquez les restrictions d'accès qu'il vous semblerait souhaitable d'appliquer.

Description et critique des DMZ :

- *DMZ 1 : zone de mise à disposition de serveurs publics accessibles depuis Internet. On y voit apparaître un serveur Web (HTTP et HTTPS) et un serveur DNS (ce dernier étant probablement utilisé pour publier le nom du serveur Web).*
- *DMZ 2 : serveur de base de données. Ce serveur de données est accédé par le serveur Web de la DMZ 1, mais également depuis le réseau local. C'est apparemment, le serveur de base de données « orienté Internet » qui contient, par exemple, le « panier » des clients d'un site Web marchand en train d'effectuer leurs achats, le catalogue des produits disponibles via Internet, les relevés de comptes à J-1, etc. Ce serveur est présent dans une DMZ spécifique pour l'isoler d'une éventuelle attaque réussie sur les machines de la DMZ 1.*
- *DMZ 3 : zone d'installation des relais utilisés pour les flux sortants du réseau local vers Internet. On y voit deux relais, un relais HTTP (squid) et un serveur DNS (BIND) très probablement utilisé en mode relais seulement. Les machines du réseau local*

utilisent le protocole de cache HTTP pour accéder à ce relais et donc aux pages d'Internet.

- *DMZ 4 : zone d'arrivée des connexions externes par modem.*
- *DMZ admin. : DMZ d'administration du firewall. Celui-ci est administré depuis cette zone par le protocole HTTPS. (Il est très probable qu'on a interdit de l'administrer depuis une autre de ses interfaces réseau.) Le firewall peut également stocker ses traces sur la machine présente dans cette DMZ en les émettant via syslog.*

*Les DMZ 1 et 2 permettent à l'entreprise de fournir un **service sur Internet** via un serveur Web.*
*La DMZ 3 offre un service **d'accès HTTP sortants** pour les postes de travail du réseau local.*
*La DMZ 4 offre un service **d'accès téléphonique** au réseau local de l'entreprise.*
*Enfin, la DMZ d'administration offre une fonction **d'administration sécurisée** de l'architecture de communication.*

- *Service Internet :*
  - *Avantages :*
    - *Protection du système de gestion des données (BD Oracle)*
    - *Possibilité d'activer plusieurs serveurs HTTP pour augmenter les performances*
  - *Inconvénients :*
    - *Difficulté d'administration.*
    - *Il est difficile de bien spécifier/contrôler les transferts d'information entre le réseau local et le serveur de base de données utilisé pour le service Internet.*
- *Accès HTTP sortants :*
  - *Avantages :*
    - *Possibilité de faire une authentification des utilisateurs au niveau du relais Squid.*
    - *Possibilité de faire du filtrage d'URL au niveau de ce relais.*
  - *Inconvénients :*
    - *Les clients du réseau local ne peuvent pas facilement utiliser d'autres protocoles que HTTP.*
    - *Les performances sont assez sensibles aux variations d'utilisation du relais Squid (et à la manière dont il est configuré : présence ou non de filtrage d'URL, etc.).*
- *Accès téléphonique :*
  - *Avantages :*
    - *Les flux transitent par le firewall : on peut faire un contrôle des protocoles, ou une écoute du réseau.*
  - *Inconvénients :*
    - *Il n'y a pas de contrôle des protocoles autorisés à entrer sur le réseau local (tous les protocoles TCP sont autorisés).*
- *Administration sécurisée :*
  - *Avantages :*
    - *Le poste d'administration du firewall est clairement identifié.*
  - *Inconvénients :*
    - *On doit créer et protéger physiquement un réseau Ethernet isolé (problème d'organisation physique, de cablage, d'accessibilité, etc.)*

Restrictions d'accès complémentaires :

- *Il est extrêmement souhaitable de limiter les protocoles TCP utilisables par les ordinateurs accédant via la serveur PPP.*
- *SQL*Net vers le DMZ 2 : Il est très souhaitable de limiter les accès aux seules adresses IP des machines ou des postes de travail du réseau local qui en ont absolument besoin.*
- *ICMP : Limiter à ICMP-echo et ICMP-reply pour la traversée du firewall*
- *DNS : Si les postes de travail sont censés utiliser le serveur DNS présent sur le réseau local, on peut limiter les accès DNS vers la DMZ 3 aux seules demandes provenant de l'adresse source du serveur DNS interne (sur le réseau local).*
- *HTTP-cache : A priori, il n'est pas nécessaire que le serveur interne accès au relais HTTP de la DMZ 3.*
- *HTTP (vers la DMZ 1) : Il est peut-être souhaitable de limiter ces accès directs aux gestionnaires du serveur Apache de la DMZ 1.*

Exercice 3

Voici 3 exemples de méthode utilisables par un utilisateur pour choisir un mot de passe :

1. Utilisation d'un mot de passe de 7 symboles choisis au hasard parmi les **36** symboles alphanumériques (en supposant que le système d'exploitation ne fait *pas* de distinction entre majuscules et minuscules).
   Exemple : `GB4F7BB`
2. Utilisation de 2 mots du dictionnaire français courant accolés ou éventuellement séparés ou suivis par des caractères spéciaux choisis parmi : `,;:!?.+-*/<>=`
   Exemples : `LUNE;MIEL` ou `ALLEZ-BLEUS!`
3. Utilisation de la première lettre d'une phrase comptant au moins douze mots. On fera aussi l'hypothèse qu'au plus deux signes de ponctuation (parmi 6 : `,;:!?.`) peuvent apparaître (n'importe où entre les douze mots) et qu'il est possible qu'au plus deux lettres soient en majuscules.
   Exemple : `Udlpld'upcamdm.` (1ère phrase ci-dessus)

Question 1 : Comparez ces méthodes en évaluant *quantitativement* le nombre de mots de passe différents associé à chaque méthode, en vous servant si besoin des informations indiquées ci-dessous. Il n'est pas indispensable de calculer de manière algébrique exacte la taille de l'espace considéré, mais essayez d'obtenir un ordre de grandeur *justifié* du nombre de mots de passe possibles (ou un minorant).

Question 2 : Ensuite, en vous appuyant sur les informations expérimentales fournies ci-dessous, calculez (en jours) la durée espérée de résistance d'un mot de passe de type 1, 2 ou 3 face à un outil d'attaque par dictionnaire[1], suite à un vol de la forme chiffrée du mot de passe[2]. Considérez : un système Unix utilisant un chiffrement DES classique, un système Windows utilisant NT LM (DES) et un système utilisant MD5.

---

[1] Une attaque consistant à essayer systématiquement les mots d'un dictionnaire ou des combinaisons simples de ces mots entre eux ou avec des symboles courants.

[2] Contenue dans /etc/passwd ou /etc/shadow sur un système Unix, ou dans la base SAM d'une machine Windows, ou transitant sur le réseau pour certaines applications (VNC, etc.).

Quelques information utiles :

- On pourra considérer qu'un lexique de français courant compte environ 4000 mots[3] (méthode 2).
- (*Pour simplifier, il est optionnel de considérer cette hypothèse.*) Pour plus de réalisme, on pourra considérer (méthode 3) qu'une phrase est composée à 50% de « mots outils » qui ne représentent que 0,5% du lexique total (soit 20 mots au lieu de 4000), mais surtout qui ne représentent probablement que 25% des lettres de l'alphabet (pour leur première lettre), soit seulement 6 lettres.
- Le plus sûr est souvent d'évaluer expérimentalement le nombre de combinaisons par seconde (c/s) qu'une machine peut essayer[4]. Voici, sur une machine tout à fait banale (PII 266 MHz) les performances obtenues par un outil librement disponible :

```
# /home/ripper/john-1.6.36/run/john -test
Benchmarking: Traditional DES [64/64 BS MMX]... DONE
Many salts:     80230 c/s real, 80230 c/s virtual

Benchmarking: FreeBSD MD5 [32/32]... DONE
Raw:    644 c/s real, 644 c/s virtual

Benchmarking: NT LM DES [64/64 BS MMX]... DONE
Raw:    493030 c/s real[5], 492046 c/s virtual

Benchmarking: OpenBSD Blowfish (x32) [32/32]... DONE
Raw:    44.7 c/s real[6], 44.8 c/s virtual

Benchmarking: NT MD4 [TridgeMD4]... DONE
Raw:    124764 c/s real, 124764 c/s virtual
```

Pour les 3 méthodes, le nombre de mots de passe possibles est le suivant :

1. $36^7 = 78\,364\,164\,096 \cong 7{,}8.10^{10}$ *possibilités*
2. *On* 4000^2 *possibilités pour le choix des 2 mots, et ensuite* $(13+1)^2$ *possibilités pour choisir, soit un des 13 symboles, soit rien du tout, entre ou après les 2 mots ; ce qui donne au total :* $4000^2 \times (13+1)^2 = 3\,136\,000\,000 \cong 3{,}1.10^9$ *possibilités.*
3. *Nous considérons tout d'abord le cas où les mots-outils ne sont pas considérés (3.A), puis celui où ils sont considérés (3.B) :*
   A. *On aura tout d'abord le choix de 12 lettres parmi l'alphabet de 26, soit 2612 possibilités. De la même manière, on aura le choix de 2 parmi les 6 symboles de ponctuation et la possibilité de ne pas avoir de symbole. Enfin, pour la position des lettres en majuscule comme pour la position des deux signes de ponctuation (ceux-ci n'étant pas consécutifs), on aura* 12×11 *possibilités. Soit au total :* $26^{12} \times (6+1)^2 \times (12 \times 11)^2 = 81\,474\,952\,902\,784\,361\,496\,576 \cong 8{,}1.10^{22}$
   B. *Dans ce cas, on peut considérer qu'un caractère sur deux du mot de passe est choisi parmi seulement 6 lettres de l'alphabet, et non 26 comme les autres. On garde les*

---

[3] Le lexique Dubois-Buyse des mots français courants (enfants entre 0 et 16 ans) compte 3726 mots.

[4] La quasi-totalité des systèmes d'exploitation utilisent un « grain de sel » (*salt*) aléatoire pour ralentir les attaques par dictionnaire (c'est une valeur concaténée au mot de passe de l'utilisateur qui multiplie immédiatement le nombre de combinaisons à essayer pour une attaque par dictionnaire). Voir : crypt(3). Le temps nécessaire pour chiffrer et comparer un mot du dictionnaire à la valeur dérobée dépend bien sûr du type d'algorithme (DES, MD5, MD4, Blowfish, etc.), mais dépend aussi fortement de la longueur du « *salt* » (10 bits, 12 bits, ou plus).

[5] Pour votre information, il y a des cas où les machines Windows n'utilisent pas de « *salt* » avec NT LM (cf : note 4). Dans ces cas, expérimentalement, on atteint plus de 20 000 000 c/s sur cette machine…

[6] Avec un « *salt* » de 128 bits…

mêmes termes que précédemment à l'exception du premier, coupé en deux ; soit au total : $\left(6^6 \times 26^6\right) \times (6+1)^2 \times (12 \times 11)^2 = 12\,305\,280\,914\,602\,131\,456 \cong 1{,}2.10^{19}$

En divisant le nombre total de mots de passe possibles par la vitesse d'essai de l'outil, on obtient alors les résultats suivants (valeurs approchées minorées).

	Cas 1	*Cas 2*	*Cas 3.A*	*Cas 3.B*
DES	*11 j*	*0,45 j (11 h)*	$1{,}1.10^{19}$ *j*	$1{,}7.10^{9}$ *j*
NT LM	*1,8 j*	*0,073 j (1,7 h)*	$1{,}9.10^{12}$ *j*	$2{,}8.10^{8}$ *j*
MD5	*1400 j*	*56 j*	$1{,}4.10^{15}$ *j*	$2{,}2.10^{11}$ *j*

En conclusion, comme on peut le voir, les méthodes 1 et 2 restent largement vulnérables à ce type d'attaque, surtout dans les cas où l'algorithme de stockage du mot de passe sur le système d'exploitation utilise des techniques « anciennes ». La méthode 3 semble fournir un meilleur niveau de sécurité, essentiellement dû à l'augmentation de la longueur du mot de passe (et au fait qu'on n'utilise pas directement les mots d'un dictionnaire).
Dans tous les cas, compte tenu de l'efficacité de cette attaque dans la pratique, il faut souligner l'importance de protéger les serveurs contenant la forme chiffrée des mots de passe (utilisée en entrée pour l'outil d'attaque) : par exemple les contrôleurs de domaine, ou les gestionnaires NIS ou LDAP.