TIW4 – Sécurité des Systèmes d'Informations

Livret d'exercices

Romuald Thion

Master 2 *Technologies de l'Information et Web (TIW)* – 2019–2020

Table des matières

Liste des exercices

Chapitre 1

Politique de sécurité et analyse de risques

1.1 Analyse de risques Ebios : généralités

Exercice 1 : Bien supports et menaces génériques (*[Age10a, Chapitres 1, 2, 4]*)

1. Classer les bien supports suivants selon les catégories : *matériels (MAT), logiciels (LOG), canaux informatiques et de téléphonie (RSX), personnes (PER), supports papier (PAP), canaux interpersonnels (CAN), locaux (LOC)* :
   — fibre optique
   — document imprimé
   — cartouche de sauvegarde
   — commutateur téléphonique
   — assistant personnel (PDA)
   — SGBD Oracle
   — discussions de couloir
   — salle de réunion
   — Linux
   — client de courrier électronique
   — poste de travail
   — salle de conférence
   — ligne téléphonique
2. Proposer des exemples d'impacts génériques : sur le fonctionnement, les humains, les biens. Quels autres impacts ne rentrent pas dans les catégories précédentes ?
3. Pour les catégories LOG et CAN précentes, proposer des menaces génériques en précisant le(s) critère(s) de sécurité concernés et les vulnérabilités exploitables. Par exemple pour PER on proposerait "Dissipation de l'activité d'une personne" par l'exploitation du temps de travail, du blocage de l'accès d'une personne ou l'exploitation d'une personne en dehors de ses prérogatives. Cette menace porte atteinte à la disponibilité de la personne et sera efficace sur les sujets à la dissipation.

Exercice 2 : Évaluation des risques (*[Age10c, p. 62]*)

L'action 4.1.1. de la méthode Ebios « Analyser les risques » consiste à mettre en évidence l'ensemble des risques qui pèsent réellement sur le périmètre de l'étude et à déterminer leur gravité et leur vraisemblance, une première fois sans tenir compte des mesures de sécurité existantes, et une seconde fois en les prenant en compte. On fait ainsi le lien entre les événements redoutés et les scénarios de menaces, c'est-à-dire entre ce que l'organisme craint et ce à quoi il est exposé.

1. Connaissant les résultats produits par les étapes précédentes de la méthode, expliquer comment identifier les risques.

1.2 Analyse de risques Ebios : cas d'étude UCBL

Description du cas La Direction du Système d'Information (DSI) de l'UCBL désire mettre en place une Politique de Sécurité du Système d'Information (PSSI). La complexité du SI impose d'utiliser une méthode pour recenser et classifier exactement ce qu'il faut sécuriser. La méthode Expression des Besoins et Identification des Objectifs de Sécurité (EBIOS) est retenue pour conduire cette étude et aboutir à la définition de la PSSI. Parmi les services de l'UCBL, notons celui des *finances*, en charge de la comptabilité, de la gestion des budgets, des marchés et des payes. Les services financiers s'appuient sur le logiciel SIFAC (Système d'Information, Financier Analytique et Comptable) hébergé sur un serveur du bâtiment Braconnier. Notons que l'activité services financiers n'est pas régulière, en effet, ces services sont particulièrement sollicités chaque fin de mois pour les payes et de façon intense aux mois de novembre et de début décembre avec la clôture de l'exercice budgétaire.

Exercice 3 : Étude du contexte ($[Age10a,\ Age10c]$)

1. Les sources de menaces au sens Ebios sont les sources non humaines (code malveillant, phénomène naturel, catastrophe naturelle ou sanitaire, activité animale, événement interne) ou humaines. Les humaines sont décomposées en interne/externe, avec/sans intention de nuire et selon leur capacité (faibles, importantes, illimitées). Indiquer quelles sont les sources de menaces qui peuvent raisonnablement être écartées du contexte de l'étude.
2. Donner des exemples de sources de menaces pour les types de sources de menaces suivants :
   1. Source humaine interne, sans intention de nuire, avec de faibles capacités ;
   2. Source humaine interne, sans intention de nuire, avec des capacités illimitées ;
   3. Source humaine externe, sans intention de nuire, avec de faibles capacités ;
   4. Source humaine externe, malveillante, avec de faibles capacités ;
   5. Événement interne.
3. Quels sont les critères traditionnels de la sécurité ? La DSI souhaite ajouter le critère de *traçabilité* dans le périmètre de son étude. Proposer une définition de ce critère et justifier l'inclusion de ce nouveau critère dans le contexte.

Exercice 4 : Détermination des objectifs de sécurité ($[Age10a,\ Age10c]$)

Dans le module d'étude des risques Ebios, les risques intolérables du service des finances sont :
— risque lié à l'arrêt de SIFAC en période de clôture ;
— risque lié l'usurpation d'identité sur la messagerie électronique ;

1. Quelles sont, en général, les différentes possibilités de traitement des risques ? Lesquelles sont envisageables dans le contexte de l'étude ?
2. Proposer des mesures de sécurité pour *réduire* les risques.

1.3 Analyse de risques Ebios : cas d'étude @RCHIMED

Description du cas La société @RCHIMED est un bureau d'ingénierie en architecture. Cette PME toulonnaise est constituée d'une douzaine de personnes.

La société @RCHIMED réalise des plans d'usines ou d'immeubles avec l'établissement préalable de devis. Pour cela, elle calcule des structures, élabore des plans techniques pour ses architectes et

propose des maquettes virtuelles pour ses clients. Le suivi des constructions est aussi assuré par le cabinet, qui met à jour les plans et calculs si des modifications sont nécessaires.

Le cabinet d'architecture bâti sa réputation grâce à des solutions architecturales originales basées sur des techniques innovantes. Cette société concourt pour de grands projets nationaux ou internationaux ; elle s'appuie pour cela sur son système informatique qui lui permet de réagir extrêmement rapidement aux appels d'offre ou aux demandes des clients.

Elle attache également une importance extrême à la qualité des documents remis et plus précisément aux maquettes virtuelles (visualisations 3D) qui permettent de donner à ses clients une idée précise et concrète de la solution proposée. Dans un contexte de rude concurrence, rapidité, précision et originalité des travaux sont des composantes essentielles de son activité.

Par ailleurs, elle a créé son site Internet sur lequel sont présentés les informations concernant la société et des exemples de devis et de maquettes virtuelles.

L'informatique de la société est reliée par un réseau wifi et le bureau d'études dispose d'un réseau local de type Ethernet. Le site Internet est hébergé sur un serveur externe. Le bureau d'étude possède 7 ordinateurs, le service commercial 2 ordinateurs portables, le service comptabilité 1 ordinateur, et le service de gestion de site Internet 1 ordinateur.

Sécurité du système d'information Il n'y a pas de principes généraux, ni de politique de sécurité, seulement les quelques règles suivantes :

- le contrôle d'accès se fait par identifiant /mot de passe ;
- principe de sauvegarde de tout fichier ;
- chaque ingénieur est responsable du fichier qu'il traite, les fichiers sont sauvegardés sur des disques USB stockés dans une armoire fermant à clé, située dans le bureau d'études ;
- parallèlement, les documents papiers sont rangés dans une armoire forte du service commercial ;
- en ce qui concerne la maintenance, un contrat a été établi avec les fournisseurs de logiciels avec intervention sous 4 heures.

Conjoncture La mise en réseau des systèmes informatiques s'est effectuée avec succès et a permis de réduire encore plus les délais de réalisation des travaux. L'entreprise doit maintenant répondre au souhait de la majorité des clients qui est de correspondre directement avec le bureau d'étude via Internet pour transmettre tous les types de documents (dossiers techniques, devis, appel d'offre, messages...).

L'entreprise a dernièrement perdu un marché : la rénovation de la mairie de Draguignan. Lors de la présentation des projets, il est apparu de curieuses similitudes entre la maquette virtuelle d'@RCHIMED et la proposition d'un concurrent de Nice. Le directeur d'@RCHIMED soupçonne une compromission du projet qu'il avait présenté. Il a maintenant des craintes sur la confidentialité de certains projets.

D'autre part, de plus en plus de contrats pour lesquels @ARCHIMED souhaite se positionner sont conditionné par la capacité du cabinet à assurer la confidentialité relative aux aspects techniques du projet. Par exemple, L'appel d'offre pour la rénovation de certaines installations de la marine nationale de l'arsenal de Toulon entre dans ce cadre.

Compte tenu de son volume et de sa disposition, la société travaille de façon très ouverte. Cependant, les experts du bureau d'études sont les seuls à pouvoir accéder aux logiciels les plus performants de conception et de maquettage. Ces experts ont par ailleurs bénéficié d'une formation à la manipulation de ces outils. Chacun est conscient de ces responsabilités financières, civiles et pénales associées à l'usage des informations qu'il manipule : dossier client, données nominatives...

Le choix d'une étude de sécurité s'impose donc pour, d'une part, déterminer les conditions qui permettent l'ouverture du système informatique vers l'extérieur et d'autre part pour déterminer les mesures de sécurité nécessaires à la protection des projets sensibles.

Exercice 5 : étude du contexte (*[Age10b, p. 15, p.18]*)

1. Donner des exemples de sources de menaces à prendre en compte dans l'étude pour les types de sources de menaces suivants :
   1. Source humaine interne, sans intention de nuire, avec de faibles capacités ;
   2. Source humaine interne, sans intention de nuire, avec des capacités illimitées ;
   3. Source humaine externe, malveillante, avec des capacités importantes ;
   4. Source humaine externe, malveillante, avec des capacités illimitée ;
   5. Source humaine externe, sans intention de nuire, avec de faibles capacités ;
   6. Événement interne.
2. Identifier 3 processus (métiers) essentiels du cabinet. Quelles sont les informations essentielles concernées ?

Exercice 6 : étude des événements redoutés (*[Age10b, p. 22]*)

Pour les activités de création de plans et de calculs de structures, évaluer les événements redoutés selon les 3 critères de sécurité disponibilité, intégrité, confidentialité. On utilisera les échelles suivantes :

— Disponibilité : *plus de 72h* $\preceq$ *entre 24 et 72h* $\preceq$ *entre 4 et 24h* $\preceq$ *moins de 4h*.
— Intégrité : *détectable* $\preceq$ *maîtrisé* $\preceq$ *intègre*.
— Confidentialité : *public* $\preceq$ *limité* $\preceq$ *réservé* $\preceq$ *privé*.
— Gravité : *négligeable* $\preceq$ *limitée* $\preceq$ *importante* $\preceq$ *critique*.

Evt.	Besoin	Source	Impacts	Gravité
Indisponibilité				
Altération				
Compromission				

Exercice 7 : étude des scénarios de menaces (*[Age10b, p. 24]*)

Détailler les sources et la vraisemblance des menaces portant sur le réseau wifi. On utilisera l'échelle de vraisemblance suivante : *minime* $\preceq$ *significative* $\preceq$ *forte* $\preceq$ *maximale*.

Menace	Source	Vraisemblance
Menaces sur le réseau wifi causant une indisponibilité		
Menaces sur le réseau wifi causant une altération		
Menaces sur le réseau wifi causant une compromission		

Exercice 8 : détermination des objectifs de sécurité (*[Age10b, p. 41 à 48]*)

Au terme de la conduite de la méthode, les risques intolérables identifiés sont :

— risque lié à l'altération d'un devis qui doit rester rigoureusement intègre ;
— risque lié à la compromission d'un devis au-delà du personnel et des partenaires ;
— risque lié à l'altération de plans ou de calculs de structures qui doivent rester rigoureusement intègres ;

— risque lié à la compromission de plans ou calculs de structures au-delà des personnels et partenaires.

Proposer des mesures de sécurité portant sur le bien support « serveurs logiciels du réseau interne » pour réduire ces risques. Quelles autres mesures fautdrait'il appliquer pour compléter les précédentes ?

1.4 Politique de sécurité

Exercice 9 : Critères non-fonctionnels des systèmes (*[GH11, ex. n° 1.1]*)

Remplir le tableau récapitulatif en identifiant pour chacune des capacités d'un système, les critères associés (*authenticité, auditabilité, intégrité, imputabilité, sûreté, exactitude, fiabilité, disponibilité, accessibilité, continuité, traçabilité, confidentialité*) et des exemples de mesures qui contribuent à les assurer.

Capacité	Critères	Mesures
Exécuter des actions		
Permettre l'accès aux entités autorisées		
Prouver les actions		

Exercice 10 : Rentabilité d'une politique de sécurité (*[AJO10, ex. n° 105]*)

Une entreprise remarque que, statistiquement, elle souffre chaque année de 5 infections virales et 3 défigurations de son site web. La remise en état après une infection coûte 2 jours de travail de l'administrateur, soit environ 2000 €. La remise en état du site web nécessite environ 500 €. On évalue la mise en place d'un produit d'antivirus ainsi qu'un système de protection pour le site web à environ 30.000 € par an.

1. À partir des données numériques précédentes, évaluer le retour sur investissement des mesures envisagées.
2. En quoi cette évaluation n'est pas adéquate, que faut'il prendre en compte d'autre ?

Exercice 11 : Niveau de sécurité (*[AJO10, ex. n° 104]*)

On appelle *security gap* la constatation que le niveau réel de sécurité d'un système d'information est toujours inférieur à celui estimé. De plus, cette différence de niveau tend à augmenter à mesure que le temps passe.

1. Donner deux raisons qui expliquent cette différence.
2. Décrire le genre de mesures à prendre pour éviter une baisse du niveau de sécurité.

Exercice 12 : Audit de sécurité (*[AJO10, ex. n° 106]*)

Une entreprise organise un appel d'offre pour faire auditer la sécurité de son réseau. Trois offres sont proposées :

1. un *audit conceptuel* : plans, schémas et configurations du réseau sont demandés. À partir de ces informations, l'expert estimera si le réseau est sûr ou non ;
2. un *scan de vulnérabilités* : à l'aide de sondes spécialisées placées à différents endroits du réseaux, l'expert découvrira automatiquement les vulnérabilités des équipements ;
3. un *test d'intrusion* : sans aucune information préalable, l'expert tentera de pénétrer le réseau de l'entreprise depuis Internet et de s'approprier des informations confidentielles.

Chaque audit est utile à sa manière. Pour chacun, décrire une situation qui en ferait l'usage approprié.

Chapitre 2

Principes de la cryptographie

2.1 Principes de la cryptographie

Exercice 13 : Principe de Kerckhoffs (*[AJO10, ex. n° 55]*)

En 1883 Augustin Kerckhoffs a établi un principe fondamental (de la cryptographie) « il faut qu'un système cryptographique n'exige pas le secret, et qu'il puisse sans inconvénient tomber entre les mains de l'ennemi ». Expliquer ce principe.

Exercice 14 : Recherche exhaustive de clefs symétriques (*[AJO10, ex. n° 60]*)

On suppose qu'une machine spécialisée met *en moyenne* 4,5 jours pour retrouver une clef DES 56 bits par recherche exhaustive. Combien de temps mettrait-elle en moyenne pour trouver une clef 40 bits ? De 112 bits ? Une clef de 56 bits dans le pire des cas ?

Exercice 15 : Chiffrement symétrique et asymétrique (*[AJO10, ex. n° 64]*)

Un groupe de *n* personnes souhaite s'échanger *deux à deux* des informations confidentielles. Les informations échangées entre deux membres ne doivent pas pouvoir être lues par les autres.

1. Dans le cas où le groupe choisisse un système de chiffrement *symétrique*, quel est le nombre minimal de clefs nécessaires ?
2. Dans le cas où le groupe choisisse un système de chiffrement *asymétrique*, quel-est le nombre minimal de couples de clefs nécessaires ?
3. Synthétiser sous forme d'un tableau les avantages et inconvénients respectifs du chiffrement *symétrique* et *asymétrique*. Le groupe choisit finalement un système hybride qui utilise la cryptographie symétrique *et* asymétrique comme PGP, pourquoi ?

Exercice 16 : Certificats x509 avec openssl

On exécute la séquence de trois commandes pour générer un certificat (norme x509) :

1. `openssl genrsa -aes256 -out my.key 32`
2. `openssl req -new -key my.key -x509 -days 3650 -out my.crt`
3. `openssl x509 -text -in my.crt`

1. Intuiter ce que font ces trois lignes du script. Dans cet exemple, qui signe la clef publique du certificat, avec quelle clef ?
2. On exécute la commande `openssl rsa -text -in my.key`. Quelle partie commune avec le certificat sera affichée ?
3. Il est fait mention de l'aes-256 dans la première ligne de commande mais plus dans l'affichage, pourquoi ? Combien de fois un mot de passe est-il demandé lors de l'exécution de ce script ?

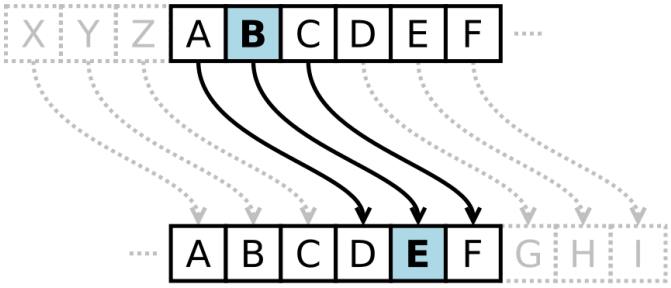
```
Certificate:
  Data:
      Version: 3 (0x2)
      Serial Number: 17264009016492929332 (0xef9614bb5b33b134)
  Signature Algorithm: sha1WithRSAEncryption
      Issuer: C=FR, ST=Rhone-Alpes, L=Lyon, O=UCBL,
              CN=Romuald THION/emailAddress=romuald.thion@univ-lyon1.fr
      Validity
          Not Before: Oct  1 15:59:57 2012 GMT
          Not After : Sep 29 15:59:57 2022 GMT
      Subject: C=FR, ST=Rhone-Alpes, L=Lyon, O=UCBL,
               CN=Romuald THION/emailAddress=romuald.thion@univ-lyon1.fr
      Subject Public Key Info:
          Public Key Algorithm: rsaEncryption
              Public-Key: (1024 bit)
              Modulus:
                  00:c5:4a:dc:49:58:d3:7f:3c:f2:94:8d:0a:d6:20:
                  80:8b:70:97:16:d2:15:73:cc:2a:be:02:e5:ec:2c:
                  ad:c3:be:39:26:7f:11:b1:99:3a:35:5d:01:2c:ab:
                  9b:3b:0a:c6:cd:3b:91:36:99:55:93:54:55:93:f8:
                  34:8e:30:13:a0:74:91:30:75:82:5d:24:23:6f:5b:
                  a5:10:7b:35:4e:98:53:99:5f:9e:df:f3:79:b6:c3:
                  b9:46:b1:46:a2:01:2a:1b:b9:fc:38:0f:a8:40:fc:
                  80:2e:aa:7e:71:dc:bb:e0:08:33:b9:19:66:4f:fd:
                  f8:ea:6f:75:d1:d4:61:ae:c5
              Exponent: 65537 (0x10001)
      X509v3 extensions:
          X509v3 Subject Key Identifier:
              AF:82:70:CA:99:74:D2:5A:62:17:73:87:F3:43:14:E5:92:61:1A:1D
          X509v3 Authority Key Identifier:
              keyid:AF:82:70:CA:99:74:D2:5A:62:17:73:87:F3:43:14:E5:92:61:1A:1D

          X509v3 Basic Constraints:
              CA:TRUE
  Signature Algorithm: sha1WithRSAEncryption
       b1:a5:6a:c6:c8:91:c3:b6:55:a6:9d:77:27:77:46:d9:ae:c5:
       e1:b8:e9:0f:cd:e0:14:e8:a5:77:1a:d5:69:17:ea:63:44:96:
       0d:32:0f:dc:d7:e9:ee:4b:da:1d:75:e9:25:5e:32:f4:3a:0e:
       21:1b:6e:c7:46:6b:ca:3a:58:fd:2f:59:45:14:62:b6:13:ba:
       be:0a:1c:b2:79:b6:76:fc:20:16:5e:88:e8:29:1d:11:86:07:
       64:ee:d1:ee:10:ea:79:14:d3:63:48:5a:c8:a8:a5:33:ae:9d:
       be:07:73:43:0b:cf:56:9d:07:09:d5:01:90:11:bb:be:07:46:
       65:71
-----BEGIN CERTIFICATE-----
[...]
-----END CERTIFICATE-----
```

2.2 Analyse d'algorithmes de chiffrement simples

Exercice 17 : Chiffrement de César

Le chiffrement de César, ou chiffrement à décalage, est un système à substitution mono-alphabétique où chaque lettre du message en clair est décalée d'un pas constant. L'image suivante issue Wikipedia illustre le procédé dans le cas d'un décalage de 3 lettres.



1. S'agit-il d'un chiffrement à flux ou à bloc ? Combien existe-t-il de secrets pour ce chiffrement ?
2. On souhaite chiffrer le texte « Attaquez, maintenant ! ». Préciser les pré-traitements à effectuer avant de chiffrer le message et calculer son chiffré avec la clef 6.
3. Expliquer la procédure de déchiffrement et déchiffrer `dwwdtxhcpdlqwhqdqw` avec la clef 3.
4. Sachant qu'un chiffré est issu d'un clair en français, expliquer comment casser le code de César en utilisant un argument sur la fréquence des lettres.
5. Déchiffrer le message `jlzabullupntl`.
6. On propose une modification du chiffre de César où l'on utilise non plus un décalage à pas constant mais une bijection quelconque de l'alphabet dans lui-même. Comparer le nombre de clefs possibles dans ce cas au précédent. L'analyse fréquentielle est-elle encore envisageable ?

Exercice 18 : Analyse du chiffrement de Vigenère

Le chiffrement de Vigenère (1523 – 1596) est un système à substitution poly-alphabétique qui étend celui de César. Ce chiffre évite qu'une lettre du clair soit toujours chiffrée de la même façon, le rendant ainsi plus résistant aux attaques statistiques telle que celle sur le chiffre de César. Son principe est le suivant : on se donne une clef et un clair sur le même alphabet, ensuite chaque caractère de la clef est utilisé pour un décalage « à la César ». On répète la clef autant de fois que nécessaire. Par exemple :

clair	a	t	t	a	q	u	e	z	m	a	i	n	t	e	n	a	n	t
clef	a	b	c	d	a	b	c	d	a	b	c	d	a	b	c	d	a	b
chiffré	a	u	v	d	q	v	g	c	m	b	k	q	t	f	p	d	n	u

1. L'augmentation artificielle par répétition d'une clef augmente-t-elle la sécurité, par exemple, `bonjourbonjourbonjour` ?
2. Chiffrer le clair « Attaquez, maintenant ! » avec la clef `crypto`.
3. Si le texte clair est suffisamment long, il est possible d'avoir des séquences de lettres qui se répètent. À quelle conditions ces séquences auront-elles les mêmes chiffrés ?
4. Si plusieurs séquences différentes sont répétées à des intervalles différents, que peut-on en déduire sur la longueur de la clef utilisée ?

5. Déterminer les longueurs possibles de la clef utilisée pour chiffrer la séquence `eiwemcgjratpdickwseiwemciiyeawg`.
6. Une fois la longueur de clef déterminée, expliquer comment utiliser l'attaque fréquentielle de l'exercice précédent.
7. Quelle serait la solidité d'un système où l'on utiliserait des clefs à usage unique de longueurs toujours supérieures au clair. Est-ce envisageable en pratique ?

Exercice 19 : Chiffrement et déchiffrement avec RSA

RSA (pour ses auteurs Rivest, Shamir et Adleman) est un système de chiffrement et de signature à clef publique dont principe est le suivant :

1. choisir p et q, deux nombres premiers distincts ;
2. calculer le « module de chiffrement » $n = p \cdot q$;
3. calculer $\varphi(n) = (p-1) \cdot (q-1)$
4. choisir e, un entier co-premier avec $\varphi(n)$, c'est-à-dire tel quel $pgcd(e, \varphi(n)) = 1$;
5. d'après le théorème de Bachet-Bézout [1], calculer d tel que $e \cdot d \equiv 1 \pmod{\varphi(n)}$;
6. le couple (e, n) forme la clef publique, le couple (d, n) forme la clef privée.

On définit ensuite la fonction de chiffrement comme $m \mapsto m^e \pmod{n}$ et celle de déchiffrement comme $c \mapsto c^d \pmod{n}$. Pour y donné, c'est le problème de trouver x tel que $y = x^e \pmod{n}$ qui est difficile : c'est le problème de l'extraction des racines modulaires.

1. Chiffrer le message 21 avec la clef publique $(103, 143)$ [2].
2. Calculer la clef privée associée.
3. Retrouver le message en clair à partir du chiffré de 21.
4. Quelles peuvent être les difficultés lors de l'implémentation de RSA ?

2.3 Authentification

Exercice 20 : Catégorie de procédés d'authentification (*[AJO10, ex. n° 81]*)

On distingue trois grandes catégories de procédés d'authentification, dénommées « je sais », « je possède » et « je suis ». Donner un exemple de système d'authentification pour chacune.

Exercice 21 : Authentification par mots de passe (*[AJO10, ex. n° 82, n° 83]*)

I : Choix et stockage des mots de passes.

1. Pourquoi stocker les hachés des mots de passes et pas les mots de passes eux-mêmes ?
2. Pourquoi l'accès aux hachés doit-il être protégé ?
3. À quelles conditions cette protection ne serait pas nécessaire ?
4. Donner des exemples d'erreurs classiques dans le choix d'un mot de passe.
5. Proposer une procédure pour choisir un bon mot de passe.

II : Dénombrement

On s'intéresse aux mots de passe comportant uniquement des lettres minuscules et des chiffres d'au plus 7 caractères (LanManager Hash). Le benchmarking de `john` affiche :

```
Benchmarking: NT LM DES [128/128 BS SSE2-16]... DONE
Raw:    18674K c/s real, 18674K c/s virtual
```

1. $a \cdot x + b \cdot y = 1$ admet des solutions si et seulement si les entiers relatifs a et b sont premiers entre eux.
2. $21^4 \equiv 1 \pmod{143}$

1. Combien y-a-t'il de mots de passes dans l'espace considéré ?
2. Estimer la durée de l'exploration complète de l'espace des mots de passes.

2.4 Protocoles cryptographiques

Exercice 22 : Protocole imparfait (*CS 361, Dr. Bill Young*[3])

Soit le protocole suivant, où A et B s'authentifient pour partager un secret K à l'aide de chiffrement asymétrique. On note $K_{x^{-1}}$ la clef *privée* de x et K_x sa clef *publique*, on suppose que $\{\{m\}_{K_{x^{-1}}}\}_{K_x} = \{\{m\}_{K_x}\}_{K_{x^{-1}}} = m$. Le protocole est le suivant :

$A \rightarrow B : \{\{K\}_{K_{a^{-1}}}\}_{K_b}$

$B \rightarrow A : \{\{K\}_{K_{b^{-1}}}\}_{K_a}$

Ce protocole est critiquement vulnérable à une attaque, laquelle ?

Exercice 23 : Vulnérabilité du protocole *Wide Mouthed Frog*

Le protocole *Wide Mouthed Frog* se joue entre deux participants A et B en s'appuyant sur un serveur S. Le protocole est le suivant où T_A (resp. T_S) est une estampille temporelle choisie par A (resp. S) :

1. $A \rightarrow S : A, \langle T_A, B, K_{AB} \rangle_{K_{AS}}$
2. $S \rightarrow B : \langle T_S, A, K_{AB} \rangle_{K_{BS}}$

Il existe une faille à ce protocole qui permet à un attaquant de mettre à jour l'estampille T_S. Pour cela, l'attaque s'appuie sur une réutilisation du second message pour ré-initier le protocole en se faisant passer pour B auprès de S.

1. Préciser quels sont les éléments K_{AB}, K_{AS} et K_{BS} qui apparaissent dans ces messages et expliquer en français à sert ce protocole.
2. Décrire les messages de l'attaque en notant $I(X)$ l'attaquant qui se fait passer pour X.
3. Expliquer l'intérêt de l'attaque et ses limites.

Exercice 24 : Protocole d'échange de clefs de Diffie-Hellman (*[DH76]*)

Le protocole d'échange de clefs de Diffie-Hellman, du nom de ses auteurs Whitfield Diffie et Martin Hellman, est une méthode par laquelle deux personnes nommées conventionnellement Alice et Bob peuvent se mettre d'accord sur un nombre (voir Wikipedia). On suppose que les participants Alice et Bob sont d'accord sur un nombre premier p, G le groupe multiplicatif des entiers modulo p et g un générateur de G (c'est-à-dire, un nombre tel que $\{g^0, g^1, \ldots, g^n\} = G$). Le protocole est le suivant :

1. Alice choisit au hasard un nombre secret a et calcule g^a
2. Bob choisit au hasard un nombre secret b et calcule g^b
3. Alice envoie g^a à Bob
4. Bob envoie g^b à Alice
5. Bob calcule $(g^a)^b$
6. Alice calcule $(g^b)^a$

1. Soit $p = 11$ et $g = 2$. Vérifier que g est bien un générateur et calculer g^a et g^b pour $a = 7$ et $b = 5$.

3. http://www.cs.utexas.edu/~byoung/cs361/syllabus361.html

2. Quel est le secret partagé par Alice et Bob à la fin du protocole ? Expliquer comment utiliser ce secret pour établir un canal de communication sécurisé.
3. On donne $g^x = 9$ et $g^y = 3$, déterminer le secret partagé par Alice et Bob.
4. Supposons un attaquant Eve qui a la capacité d'écouter tous les messages du réseau mais d'en écrire aucun (dit *passive eavesdropper*). De quels messages Eve a-t-elle connaissance ? Quelle est l'hypothèse sur laquelle repose la sécurité du protocole ? Quelle serait la sécurité du protocole si on pouvait résoudre efficacement le problème de la question précédente ?
5. Supposons maintenant qu'Eve a aussi la capacité d'écrire (dit *forger*) des messages sur le réseau (dit *active eavesdropper*). Le protocole de Diffie-Hellman n'est *pas* résistant à cet attaquant, décrire l'attaque en précisant ce qu'Alice, Bob et Eve connaissent.
6. Proposer une variante du protocole de Diffie-Hellman pour établir un secret partagé par trois parties Alice, Bob et Carole. Combien de messages sont nécessaires ?

Exercice 25 : Protocole d'authentification ($[AJO10,\ ex.\ n^o\ 92]$)

On conçoit un protocole d'authentification simple où A désire communiquer avec B en se reposant sur une Tierce Partie (TP). K_A est la clef secrète partagée entre A et TP. K_B est la clef secrète partagée entre B et TP.

1. A crée une clef de session $K_{A,B}$ puis envoie : son identité en clair, l'identité de B et $K_{A,B}$ chiffrés avec K_A ;
2. TP déchiffre le message puis communique l'identité de A et $K_{A,B}$ chiffrés avec K_B ;
3. A et B peuvent alors communiquer ensemble en utilisant $K_{A,B}$.

1. Représenter (graphiquement) les échanges de ce protocole.
2. Pourquoi un pirate ne peut pas se faire passer pout A auprès de TP ?
3. Pourquoi B est-il certain que le message reçu provient de TP ?
4. À quelle attaque ce protocole est-il vulnérable ?
5. Proposer une correction qui n'augmente pas le nombre de messages.

Exercice 26 : Andrew Secure RPC

Considère le protocole suivant appelé *Andrew Secure RPC* :

1. $A \to B : A, \{N_a\}_{K_{ab}}$
2. $B \to A : \{N_a + 1, N_b\}_{K_{ab}}$
3. $A \to B : \{N_b + 1\}_{K_{ab}}$
4. $B \to A : \{K'_{ab}, N'_b\}_{K_{ab}}$

1. Qu'est ce que désignent K_{ab}, N_a et N_b dans ce protocole ?
2. Expliquez à quoi sert ce protocole.
3. La quatrième étape de ce protocole est vulnérable à une attaque par rejeu où un attaquant peut écouter, bloquer, forger des messages et usurper l'identité d'un tiers. Expliquez le problème et donnez une séquence d'attaque en notant $I(X)$ l'attaquant I qui se fait passer pour X.

Exercice 27 : Déploiement de PGP ($[AJO10,\ ex.\ n^o\ 98]$)

On rappelle que PGP (ou GPG) est un logiciel/protocole de chiffrement cryptographique qui permet de garantir la confidentialité et l'authentification. Il est souvent utilisé pour les courriels. Pour chaque échange, PGP tire au hasard une clef de session symétrique, cette clef sert à chiffrer

le contenu à échanger et est transmise de façon sécurisée aux destinataires à l'aide de leurs clefs publiques.

Une entreprise déploie PGP pour sa messagerie. Alice vient de générer une paire de clefs asymétriques sur son PC.

1. Représenter graphiquement les échanges.
2. Quel moyen permet d'éviter que n'importe qui lise la clef privée d'Alice sur son disque dur ?
3. Bob vient de générer son couple de clefs. Quelle démarche Alice et Bob doivent-ils entreprendre pour communiquer de façon sûre via PGP ?
4. L'étape précédente a été réalisée. Alice chiffre un message pour Bob mais ne le signe pas. PGP ne lui demande aucun mot de passe, est-ce normal ?
5. Comment sont chiffrés les messages envoyés à plusieurs destinataires ?
6. Alice chiffre son répertoire personnel avec PGP et stocke cette image chiffrée sur bande. À quel risque s'expose-t-elle si son disque dur tombe en panne ?

Exercice 28 : Serveur de clef Kerberos ($[AJO10,\ ex.\ n^o\ 89,\ n^o\ 90,\ n^o\ 91]$)

Il y a quatre participants dans le protocole Kerberos : le *serveur* (S), le *client* (C), le *serveur d'authentification* (AS) et le *serveur de tickets* (TGS). Le couple formé par le serveur d'authentification et le serveur de tickets est appelé *centre de distribution des clefs* ($KDC = AS + TGS$). Les échanges entre ces participants sont les suivants :

$C \to AS$: $[C, TGS]$

$AS \to C$: $[\{K_{C,TGS}\}_{K_C}, \{T_{C,TGS}\}_{K_{TGS}}]$, où $T_{C,TGS} = [C, \mathit{validity}, K_{C,TGS}]$

$C \to TGS$: $[S, \{T_{C,TGS}\}_{K_{TGS}}, \{A_C\}_{K_{C,TGS}}]$, où $A_C = [C, \mathit{timestamp}]$

$TGS \to C$: $[\{K_{C,S}\}_{K_{C,TGS}}, \{T_{C,S}\}_{K_S}]$, où $T_{C,S} = [C, \mathit{validity'}]$

$C \to S$: $[\{A_C\}_{K_{C,S}}, \{T_{C,S}\}_{K_S}]$

Dans lesquels :
— $K_{C,TGS}$ et $K_{C,S}$ sont des clefs symétriques de session ;
— K_C est la clef (symétrique) de C ;
— K_{TGS} est la clef (symétrique) du TGS ;
— K_S est la clef (symétrique) de S ;
— $T_{C,TGS}$ est le TGT fourni par l'AS à présenter au TGS ;
— $T_{C,S}$ est le ST fourni par le TGS à présenter à S ;

1. Dans le protocole Kerberos, un utilisateur n'a pas à s'authentifier auprès de l'AS à chaque fois qu'il désire utiliser un service. Pourquoi ?
2. Donner un avantage et un inconvénient (en terme de sécurité) de cette caractéristique.
3. L'authentification initiale via K_C utilise le chiffrement symétrique, c'est typiquement un haché du mot de passe de C. AS possède donc une liste des hachés de mots de passe. Si on arrive à compromettre cette liste, peut-on se faire passer pour un client ? Comment corriger ce problème ?
4. Un authentificateur A_C est joint à la demande de service dans Kerberos. Comment S peut vérifier que A_C a bien été crée par C ?

2.5 Cas d'étude : sauvegarde de clef pour un dossier médical

On s'intéresse à un dossier médical personnel partagé reparti et sécurisé, stocké sur un dispositif portable appelé *token*. Ce *token* est une clef USB de 16Go à laquelle est intégrée un petit processeur et un coprocesseur cryptographique, capable d'effectuer du (dé)chiffrement AES de façon efficace. En plus de la mémoire RAM usuelle, le *token* dispose d'une petite zone de mémoire sécurisée, dans

laquelle sont stockées les clefs utilisées par le coprocesseur. Les 16Go de mémoire de masse ne sont pas sécurisés (par exemple, c'est une carte SD). Il faut donc chiffrer les données que l'on y écrit pour assurer la confidentialité en cas de perte du *token*.

Un *token* peut stocker des données de son propriétaire mais aussi des données d'autres personnes. C'est typiquement le cas des *token* des professionnels de santé (pour les dossier des patients qu'ils traitent), mais aussi pour les patients eux-mêmes (on pourrait avoir un *token* pour toute une famille, ou stocker une partie du dossier d'un proche).

Dans ce système de dossier médical, chaque participant A dispose d'une paire RSA (K_A, K_A^{-1}) où K_A est la partie privée et K_A^{-1} la partie publique. L'autorité centrale C (disons, la sécurité sociale) distribue à chaque participant A un certificat C_A qui est stocké dans son *token*. Les participants sont de deux types différents : soit de type *patient* (PA), soit de type *professionnel de santé* (PS). C'est C qui définit le type de chaque participant. Le *token* est utilisé par les professionnels et les patients pour s'authentifier les uns les autres, en plus de la fonctionnalité de stockage du dossier médical. Chaque *token* stocke, en mémoire sécurisée, une clef AES 256 bits notée M_A qui est appelée *clef maître*. La clef M_A permet de chiffrer K_A. Cette clef maître permet aussi de chiffrer le dossier médical stocké dans la mémoire de masse.

Exercice 29 : Sauvegarde de la clef maître d'un *token*

On souhaite définir un protocole de sauvegarde des données du *token* en utilisant le schéma de partage de clé secrète de Shamir. Dans ce protocole, un patient A va envoyer l'intégralité de son dossier chiffré sur un serveur dans le cloud. Pour la sauvegarde confidentielle de M_A, qui permettra de recouvrir un dossier stocké dans le cloud, on souhaite faire en sorte qu'il faut le concours d'au moins un professionnel de santé et de deux personnes choisies par A pour arriver à retrouver intégralement la clef M_A. Le texte suivant est extrait de Wikipedia [4].

> *Le partage de clé secrète de Shamir (*Shamir's Secret Sharing*) est un algorithme de cryptographie. C'est une forme de partage de secret, où un secret est divisé en parties, donnant à chaque participant sa propre clé partagée, où certaines des parties ou l'ensemble d'entre elles sont nécessaires afin de reconstruire le secret.*
>
> *Formellement, notre objectif est de diviser certaines données D (par exemple, la combinaison du coffre) en n pièces $D_1, \dots, D_n$ de telle sorte que :*
> *— la connaissance de k ou plus D_i pièces rend D facilement calculable.*
> *— la connaissance de $k-1$ ou moins D_i pièces rend D complètement indéterminée.*
> *Ce régime est appelé schéma de seuil $(k; n)$. Si $k = n$ alors tous les participants sont nécessaires pour reconstituer le secret. L'idée essentielle d'Adi Shamir est que 2 points sont suffisants pour définir une ligne, 3 points suffisent à définir une parabole, 4 points pour définir une courbe cubique, etc. Autrement dit, il faut k points pour définir un polynôme de degré $k-1$.*
> *Supposons que nous voulons utiliser un schéma de seuil $(k; n)$ pour partager notre secret S, que l'on suppose, sans perte de généralité, être un élément dans un corps fini F de cardinalité P avec P un nombre premier, $0 < k \leq n < P$ et $S < P$. Les étapes du protocole sont les suivantes :*
>
> 1. *Choisir au hasard $(k-1)$ coefficients $a_1, \dots, a_{k-1}$ dans F, et poser $a_0 = S$.*
> 2. *Construire le polynôme $f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \dots + a_{k-1} x^{k-1}$.*
> 3. *Choisir au hasard n valeurs de F, par exemple $v_1, \dots, v_n$ et distribuer un couple $(v_i, f(v_i))$ à chacun des n participants.*
>
> *Étant donné un sous-ensemble de k de ces couples, nous pouvons trouver les coefficients du polynôme f à l'aide de l'interpolation polynomiale, le secret étant le terme constant $a_0 = f(0)$.*

4. https://fr.wikipedia.org/wiki/Partage_de_cl%C3%A9_secr%C3%A8te_de_Shamir

1. On considère le cas du schéma de seuil $(2, n)$ avec $P = 127$. On connaît les couples $(1, 106)$ et $(4, 44)$. Retrouver le secret S [5].
2. Pourquoi tirer une clef maître plutôt que chiffrer directement le dossier médical avec la clef RSA ?
3. Définir ce que contient C_A en utilisant la notation usuelle [6].
4. Le protocole de sauvegarde que l'on souhaite concevoir garantit-il la confidentialité, l'intégralité ou la disponibilité du dossier médical ? Justifier votre réponse pour chaque critère en une phrase maximum.
5. Lorsqu'un professionnel souhaite écrire des données sur un dossier patient, les *tokens* doivent d'abord vérifier qu'ils communiquent bien entre les bonnes personnes grâce aux certificats. Donner les étapes d'un protocole d'authentification mutuelle à clef publique en utilisant la notation usuelle. Penser à vérifier le type des participants.
6. On remarque que l'on peut combiner plusieurs itérations du système de Shamir pour faire en sorte que *au moins un professionnel de santé* et *au moins deux personnes choisies* soient nécessaires à la récupération de M_A. Expliquer comment faire.
7. Un utilisateur perd son *token*. Expliquer quelles sont les étapes à suivre pour qu'il retrouve l'intégralité de son dossier.
8. Avec $P = 127$, quelle est la taille maximum des clefs que l'on peut espérer partager ? On désire sauvegarder M_A. Quelle valeur de P doit-on choisir ?
9. En fait ce n'est pas M_A qui est utilisé pour chiffrer le dossier médical, mais une clef dérivée de M_A. Donner une méthode pour dériver n clefs différentes à partir de M_A.

5. Pour diviser a par b modulo P, on va chercher le plus petit entier congru à a modulo P divisible par b. Par exemple pour $a = 65$ et $b = 3$ on obtient $(65 \div 3) \mod 127 = ((65 + 127) \div 3) \mod 127 = (192 \div 3) \mod 127 = 64 \mod 127 = 64$. Pour la soustraction, c'est similaire : $3 - 65 \mod 127 = (127 + 3) - 65 \mod 127 = 130 - 65 \mod 127 = 65$.

6. Pour rappel, on note $\{\langle x, h(y)\rangle\}_{K_A^{-1}}$ la concaténation d'un message x avec le haché d'un message y, le tout chiffré avec la clef publique de A.

Chapitre 3

Exploitation de vulnérabilités logicielles

3.1 Bonnes pratiques de développement logiciel

Exercice 30 : Authentification en PHP

Un étudiant de licence a développé une application de gestion. Voici un extrait du code qu'il a utilisé pour l'authentification des utilisateurs de l'application.

```
1  $username = $_GET['username'];
2  $password = $_GET['password'];
3  $sqluser = mysql_query("SELECT␣COUNT(*)␣FROM␣users␣WHERE␣username␣=␣'".$username.
       "'");
4
5  $countuser = mysql_fetch_row($sqluser);
6      if ($countuser[0] == 0) {
7          $errormsg = "Your␣Username␣is␣incorrect.␣Please␣try␣again";
8      } else {
9          $sqlpass = mysql_query("SELECT␣COUNT(*)␣FROM␣users␣WHERE␣username␣=␣'".
               $username."'␣AND␣password␣=␣'".$password."'");
10         $countpass = mysql_fetch_row($sqlpass);
11         if ($countpass[0] == 0) {
12             $errormsg = "Your␣Password␣is␣incorrect.␣Please␣try␣again";
13         } else {
14             $page = $_GET['page'];
15             if (file_exists('pages/'.$page.'.php')) {
16                 include('pages/'.$page.'.php');
17                 ...
18             }
19         }
20     }
```

1. Identifier différentes failles de sécurité du code de l'authentification.
2. Quelles mesures correctives suggérer ?

Exercice 31 : Bonnes et mauvaises pratique du PhP

Une page PhP vous est fournie ci-dessous :

```
1  <?php require_once 'header.php' ?>
2
3  <h1>Free online tools</h1>
4
5  <div class="container␣background-white">
6    <!-- Nmap Box -->
7    <div class="col-md-6␣col-md-offset-3␣col-sm-offset-3">
8        <form class="" method="POST" action="">
```

```
 9            <div align="center" class="error">
10        <?php if(isset($error)) echo $error; ?>
11            </div>
12            <div class="login-header margin-bottom-30">
13                <h2>Enter an IP Address</h2>
14            </div>
15            <div class="input-group">
16                <span class="input-group-addon">
17                    <i class="fa fa-eye"></i>
18                </span>
19                <input name="ip" placeholder="ip" class="form-control" type="text">
20            </div>
21            <div class="row">
22                <div class="col-md-6">
23                    <input type="submit" class="btn btn-primary pull-right" name="
                          submit" value="Try!" />
24                </div>
25            </div>
26            <hr>
27            <h4>Nmap, the best network scanner of the world...</h4>
28            <p>
29        </form>
30    </div>
31    <!-- End Nmap Box -->
32 </div>
33
34 <?php
35    if(isset($_POST["submit"]))
36    {
37        if(empty($_POST['ip']))
38            $error = "IP field is required.";
39        else
40        {
41            $ip=$_POST['ip'];
42            echo "<pre>";
43            system("nmap -A ".$ip);
44            echo "</pre>";
45        }
46    }
47    require_once ('footer.php');
48 ?>
```

1. Expliquez quelle la fonctionnalité proposée par cette page.
2. Identifiez ses problèmes de sécurité et leurs impacts potentiels.
3. Que proposez-vous comme solution ?

Exercice 32 : Conseils pour du code sûr (*Michael Howard & Steve Lipner*)

Dans le chapitre 11 *Secure Coding Policies* de l'ouvrage *The Security Development Lifecycle : SDL : A Process for Developing Demonstrably More Secure Software*, Michael Howard & Steve Lipner, on trouve les bonnes pratiques suivantes pour l'écriture de code sûr, expliquer et justifier chacune de ces pratiques :

1. *use the latest compiler and supporting tool versions* ;
2. *use defenses added by the compiler* ;
3. *use source-code analysis tools* ;
4. *do not use banned functions*.

Exercice 33 : Guide de style pour du code robuste (*Matt Bishop & Chip Elliott*)

L'objectif de l'exercice est d'arriver, en critiquant un code C, à produire des recommandations issues de *Robust Programming by Example*, Matt Bishop & Chip Elliott, sur les bonnes pratiques pour écrire du code robuste.

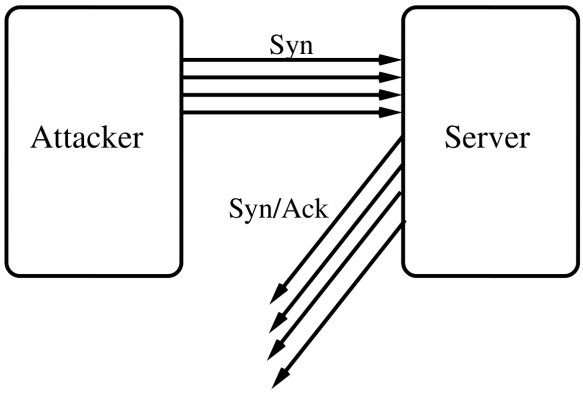
1. Que font les appels `qmanage(&qptr,85,1)`, `qmanage(&qptr,1,85)` et `qmanage(&qptr,0,85)`. Quelle(s) recommandation(s) suggérer concernant les paramètres des fonctions (en C) ?
2. Que se passe-t'il si `qptr` ou `*qptr` est `0` dans un appel à `qmanage(&qptr,0,42)`. Que se passe-t'il lorsque cet appel est passé plusieurs fois ? Quelle(s) recommandation(s) supplémentaire(s) suggérer concernant les paramètres ?
   Pour une référence en ligne exhaustive et de première qualité sur la programmation C/C++ sûre, voir `http://www.securecoding.cert.org/`. Pour un exemple de vulnérabilité causée par une double libération de la mémoire, voir par exemple `http://www.cert.org/advisories/CA-2002-07.html`

```
1  typedef struct queue {
2    int * que ;                          /* the actual array of queue
         elements */
3    int head ;                           /* head index in que of the queue
         */
4    int count ;                          /* number of elements in queue */
5    int size ;                           /* max number of elements in queue
         */
6  } QUEUE ;
7
8  void qmanage ( QUEUE ** , int , int );  /* create , delete queue */
9  void qputon ( QUEUE * , int );          /* add to queue */
10 void qtakeoff ( QUEUE * , int *);       /* remove from queue */
11
12 void qmanage ( QUEUE ** qptr , int flag , int size ){
13   if (flag) {                          /* allocate a new queue */
14     (* qptr) = malloc ( sizeof ( QUEUE ));
15     (* qptr)->head  = (* qptr )-> count = 0;
16     (* qptr)->que   = malloc ( size * sizeof ( int ));
17     (* qptr)->size  = size ;
18   } else {                             /* delete the current queue */
19     ( void ) free ((* qptr)->que);
20     ( void ) free (* qptr);
21   }
22 }
```

3.2 Reconnaissance et réseaux

Exercice 34 : `SYN` *flooding*

Le principe du *SYN flooding* consiste à submerger un serveur de requêtes `TCP` qui seront volontairement laissées dans un état semi-ouvert (plus précisément l'état `SYN_RCVD` dans lequel est un serveur qui a reçu un `SYN` mais attend toujours le `ACK` correspondant) afin de consommer un maximum de ressources sur la cible.



1. Que se passe-t'il lorsque un client essaie de se connecter légitime au serveur victime de cette attaque ?
2. Le problème du *SYN flooding* est inhérent à la façon dont une connexion `TCP` est établie. Comment réduire voire supprimer cette vulnérabilité ?
3. Une attaque de *SYN flooding distribué* (*Distributed Deny of Service*, DDoS) consiste à lancer des attaques *SYN flooding* simultanées depuis de multiples sources. Comment s'en protéger ?

Exercice 35 : Analyse de ports avec `nmap`

Les traces A & B données en annexes sont des extraits obtenus avec l'outil `nmap`. Cet outil permet de créer des paquets TCP/IP sur mesure permettant de déterminer quels sont les ports La trace A a été obtenue avec l'interface `eth0` et la trace B avec l'interface `wlan0` d'une même machine.

1. Quelle est l'IP de la machine cible et la liste des ports scannés.
2. Quelles sont les IPs de la machine qui exécute `nmap` et le port source utilisé. Pourquoi cette valeur ?
3. En quoi consiste la méthode de scan utilisée ?
4. Dans quels états sont les ports scannés (ouvert, filtré ou fermé) dans la trace A ?
5. Dans quels états sont les ports scannés (ouvert, filtré ou fermé) dans la trace B ?
6. Expliquer la différence de résultat entre les deux traces.

3.3 Étude d'exploits

Exercice 36 : Vulnérabilité logicielle *directory traversal*

On donne l'extrait de code PHP suivant qui est vulnérable à une attaque dite *directory traversal*.

```
1 <?php
2 $template = 'red.php';
3 if (isset($_COOKIE['TEMPLATE']))
4     $template = $_COOKIE['TEMPLATE'];
5 include ("/home/users/phpguru/templates/" . $template);
6 ?>
```

1. Expliquer quelle est la vulnérabilité de ce code.
2. Étant donné l'exemple suivant où un cookie contenant `XXX` pour la clef `TEMPLATE` est transmis au serveur, proposer un exemple d'exploitation de la vulnérabilité.

```
1 GET /vulnerable.php HTTP/1.0
2 Cookie: TEMPLATE=XXX
```

3. Proposer différentes mesures de protection pour limiter ces attaques au niveau :
   — du traitement des paramètres dans le code PHP ;
   — de la configuration dans l'OS hôte.

Exercice 37 : La faille CVE-2011-4029 (*Magazine MISC numéro 60*)

On s'intéresse à la vulnérabilité Xorg publiée fin 2011 sous l'identifiant CVE-2011-4029. Cette dernière exploite une « condition de concurrence » (en anglais *race condition*) de Xorg qui permet à un utilisateur local de positionner les droits en lecture sur n'importe quel fichier du système. La sortie console suivante montre une trace des appels système exécutés pour créer ce verrou lors du lancement d'un serveur X sur le display « :1 ».

```
# strace X:1
open("/tmp/.tX1-lock", O_WRONLY|O_CREAT|O_EXCL, 0644) = 0
write(0, "     20093\n", 11)            = 11
chmod("/tmp/.tX1-lock", 0444)           = 0
close(0)                                = 0
link("/tmp/.tX1-lock", "/tmp/.X1-lock") = 0
unlink("/tmp/.tX1-lock")                = 0
```

La trace est composée des étapes suivantes :

1. Ouverture d'un fichier verrou temporaire (/tmp/.tX1-lock)
2. Écriture de l'identifiant du processus (PID) dans ce nouveau fichier
3. Positionnement des droits en lecture pour tous les utilisateurs
4. Fermeture du fichier temporaire
5. Création d'un lien physique avec le véritable nom du fichier verrou (/tmp/.X1-lock)
6. Suppression du fichier verrou temporaire (/tmp/.tX1-lock)

C'est dans la manière dont est manipulé le fichier verrou temporaire (étapes 1 à 4), lors du lancement d'un serveur X, que la faille réside. Voici un extrait de la fonction `LockServer()` du fichier `os/utils.c` :

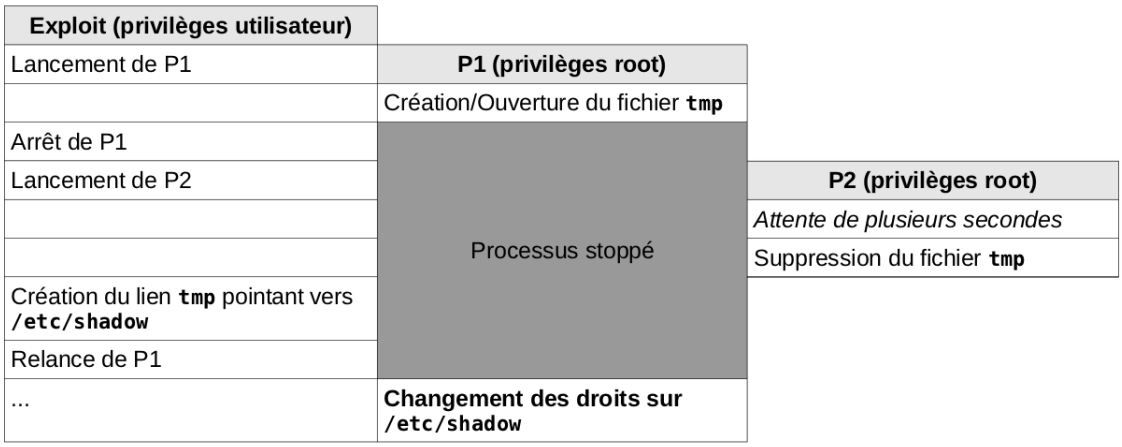
```
294   do {
295     i++;
296     lfd = open(tmp, O_CREAT | O_EXCL | O_WRONLY, 0644);
297     if (lfd < 0)
298        sleep(2);
299     else
300        break;
301   } while (i < 3);
...
314    if (lfd < 0)
315      FatalError("Could not create lock file in %s\n", tmp);
316    (void) sprintf(pid_str, "%10ld\n", (long)getpid());
317    (void) write(lfd, pid_str, 11);
318    (void) chmod(tmp, 0444);
319    (void) close(lfd);
...
328     haslock = (link(tmp,LockFile) == 0);
329     if (haslock) {
...
333       break;
334     }
335     else {
336       /*
337        * Read the pid from the existing file
338        */
339       lfd = open(LockFile, O_RDONLY);
340       if (lfd < 0) {
341         unlink(tmp);
342         FatalError("Can't read lock file %s\n", LockFile);
343       }
```

La fonction `open()` permet la création du fichier `tmp` : en cas de réussite, `open()` assure qu'aucun fichier du même nom n'était présent sur le disque et renvoie un *descripteur de fichier*. A l'inverse de `fchmod()`, la fonction `chmod()` opère sur un *nom* de fichier et non sur un *descripteur* de fichier.

L'exploit de type « Time-Of-Check-To-Time-Of-Use » de la faille CVE-2011-4029, consiste, entre les lignes lignes 296 et 317, à supprimer puis remplacer le fichier *tmp* par un lien symbolique pointant vers un fichier arbitraire, afin que le changement de droits de la ligne 318 soit effectué sur le fichier de notre choix.

La difficulté de l'exploit est de réussir à intercaler précisément ces actions dans l'exécution de `LockServer()`. Le principe général de l'exploit, résumé par la figure 1, consiste :

- à exécuter une première instance (P1) de Xorg contrôlée via les signaux `SIGSTOP` et `SIGCONT` qui permettent respectivement de mettre en pause un processus et de le relancer là ou il s'était arrêté. L'arrêt de la première instance d'Xorg doit se faire immédiatement après la création du fichier temporaire (ligne 296) ;
- à exécuter une deuxième instance (P2) qui supprimera le fichier `tmp` pendant que (P1) est stoppé ;
- à scruter efficacement le système de fichier. Pour cela, l'API `Inotify` permet de reporter à une application tout changement sur un système de fichier au lieu de guetter l'apparition d'un fichier en testant sa présence avec une boucle `while()`.



(Figure 1) Fonctionnement de l'exploit

1. Expliquer le problème de la fonction `LockServer()`.
2. Comment et pourquoi (P2) supprimera le lien physique (que l'on assimilera à un fichier) vers `LockFile` ?
3. Quel est l'intérêt de `Inotify` ? Qu'adviendrait-il si on utilisait une technique naïve de boucle `while()` pour scruter l'activité dans `/tmp` ?
4. Pourquoi le lancement de l'instance (P1) dans l'exploit se fait avec la plus basse priorité ?
5. Quelle correction pourrait-on apporter pour supprimer cette faille ?
6. Dans le *proof of concept* de l'exploit, le fichier sur lequel est positionné `0444` est `/etc/shadow`. Donner des exemples d'impacts que l'exploitation de cette technique permettrait d'avoir. Quel autres fichiers pourrait-on vouloir utiliser ?

Exercice 38 : Analyse du bulletin MS10-092

Bulletin MS10-092 : *This security update resolves a publicly disclosed vulnerability in Windows Task Scheduler. The vulnerability could allow elevation of privilege if an attacker logged on to an affected system and ran a specially crafted application. An attacker must have valid logon credentials and be able to log on locally to exploit this vulnerability. The vulnerability could not be exploited remotely or by anonymous users. [...]*

An elevation of privilege vulnerability exists in the way that the Windows Task Scheduler improperly validates whether scheduled tasks run within the intended security context. An attacker who successfully exploited this vulnerability could run arbitrary code in the security context of the local system. An attacker could then install programs ; view, change, or delete data ; or create new accounts with full user rights.

Les jobs du Task Scheduler sont décrits par des fichiers xml de la forme suivante. Notons qu'il est possible d'ajouter des commentaires dans ce fichier avec les balises `<!-- comment -->`. Le dossier où sont stockés ces fichiers est lisible par LocalSystem et les administrateurs locaux. Un utilisateur (sauf guest) peut écrire dans ce dossier. Pour protéger l'intégrité des commandes (qui sont exécutée par le Task Scheduler qui est un utilisateur privilégié), un checksum est calculé à la création de la tâche. Quand le job est lancé, le checksum est recalculé et comparé à celui enregistré avant d'être exécuté. Un algorithme de *cyclic redundancy check* (CRC) est utilisé pour cela (en l'espèce, CRC32).

```
<Principals>
  <Principal id="LocalSystem">
    <UserId>S-1-5-18</UserId>
    <RunLevel>HighestAvailable</RunLevel>
  </Principal>
</Principals>
<Actions Context="LocalSystem">
  <Exec>
    <Command>C:\WINDOWS\NOTEPAD.EXE</Command>
    <Arguments></Arguments>
  </Exec>
</Actions>
```

La description de l'exploit est la suivante :

1. *Create a job that will be run under the current user account with the least available privileges ;*
2. *Read the task configuration file corresponding to the task created at step 1 and calculate its CRC32 checksum ;*
3. *Modify the task configuration file corresponding to the task created at step 1 so that it matches the same check sum as the original file and set the following properties :*
   (a) *Run the task.Principal Id=LocalSystem (principal for the task that provides security credentials) ;*
   (b) *Run the task.UserId=S-1-5-18 (SID of the LocalSystem) ;*
   (c) *Run the task.RunLevel=HighestAvailable (run with the highest available privileges) ;*
   (d) *Run the task.Actions Context=LocalSystem (security context under which the actions of the task are performed) ;*
4. *Run the task.*

1. Quel est le problème qui rend l'étape 3 de cet exploit possible ? De quelle forme d'attaque cryptographique s'agit-il ? Quelle solution proposeriez-vous ?
2. Dans quelle catégorie de la classification SANS classeriez-vous cet exploit ? (à voir après le cours sur les vulnérabilités).

Chapitre 4

Gestion des autorisations

4.1 Modèles de contrôle d'accès

On note S l'ensemble des sujets du système, O celui de ses objets et A les actions/privilèges disponibles sur les objets.

Exercice 39 : Modèle Harrison-Ruzo-Ullman

Dans le modèle HRU (Harrison-Ruzo-Ullman), un état du système est représenté par un triplet (S, O, M), où $M : S \times O \to \wp(A)$ est une matrice de contrôle d'accès qui à chaque paire $(s, o) \in S \times O$ fait correspondre les privilèges de s sur o.

	Fichier1	*Fichier2*	*Fichier3*	*Fichier4*
Alice (a)	owns	r	r	
Bob (b)	r	owns	rw	owns
Charly (c)	r	rw	owns	rw
Denise (d)			r	rw

Le modèle HRU définit un ensemble de 6 opérations primitives pour modifier un état (S, O, M) :
— `enter` a `into` $M(s, o)$
— `delete` a `into` $M(s, o)$
— `create subject` s
— `delete subject` s
— `create object` o
— `delete object` o

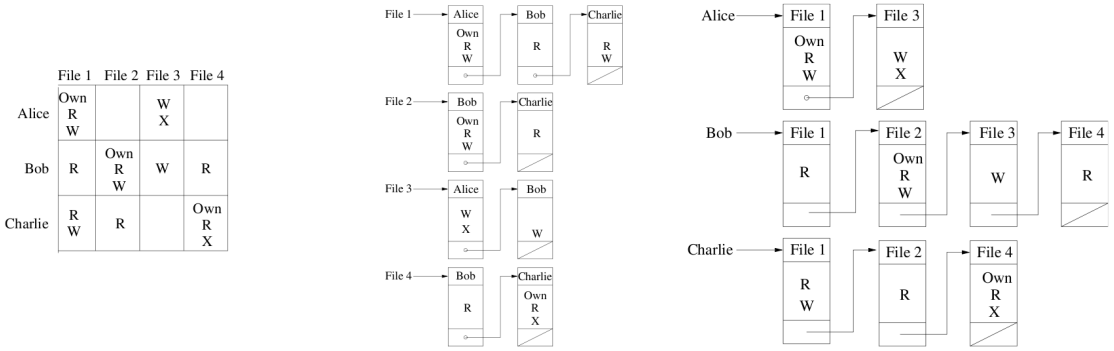
1. On note $(S, O, M) \vdash_c (S', O', M')$ le changement d'état associé à la commande c. Donner, formellement ou en pseudo-code, la définition des quatre premières opérations primitives.
2. On considère comme état initial la matrice donnée en exemple. Donner l'état obtenu après l'exécution de la séquence d'opérations suivante :
   1. `enter` w `into` $M(a, 3)$
   2. `create object` 5
   3. `create object` 5
   4. `enter` *owns* `into` $M(e, 5)$
   5. `delete object` 2
3. À partir des opérations et des structures de contrôle de base, on peut construire des opérations complexes :

command *confer*.*read*($s1, s2, o$)
if *owns* $\in M(s1, o)$ **then**
`enter` r `into` $M(s2, o)$
end if

Que fait cette commande ?

4. Similairement, définir les commandes
— *confer*.*write*($s1, s2, o$)
— *revoke*.*read*($s1, s2, o$)
5. Définir une commande *create*.*file*(s, o) qui crée un fichier o dont s est le propriétaire de o et sur lequel tous les utilisateurs peuvent lire le fichier nouvellement crée.
6. La matrice des droits n'est généralement pas gérée telle que par les systèmes. On lui préfère une représentation sous forme de listes chaînées, où les chaînes sont indexées soit par les objets (*access control list*), soit par les sujets (*capabilities list*).

Access Matrix **AC List (ACL)** **Capabilities List**



Les systèmes de gestion de fichier privilégient la forme *access control list*. Pourquoi ? Quelles applications seraient susceptibles de privilégier la forme *capabilities list* ?

Exercice 40 : Modèles mandataires à niveaux

Les modèles mandataires à niveaux font l'hypothèse que les niveaux de sécurité disposent d'une structure *d'ordre partiel*, c'est-à-dire que l'ensemble L des niveaux est muni d'une relation $\leq_L \subseteq L \times L$ qui soit transitive, réflexive et antisymétrique. À chaque sujet et à chaque objet est associé un unique niveau de sécurité, formellement $cl : S \to L$ et $la : O \to L$.

1. Calculer la matrice d'accès $M \subseteq \{a, b, c\} \times \{r, w\} \times \{1, 2, 3, 4\}$ obtenue avec :
— $L = \{pub, conf, secret, top\}$ totalement ordonné $pub \leq_L conf \leq_L secret \leq_L top$
— la fonction cl telle que $cl(a) = public$, $cl(b) = secret$, $cl(c) = confidentiel$
— la fonction la telle que $la(1) = topsecret$, $la(2) = public$, $la(3) = secret$, $la(4) = public$
2. Montrer que seuls les utilisateurs qui ont exactement le même niveau qu'un objet peuvent lire et écrire dessus.
3. Montrer que certaines matrices de contrôle de droits ne peuvent pas être exprimées par un modèle à niveaux. (Indice : utiliser la réponse à la question précédente).
4. À partir de deux ordres partiels $(E, \leq_E)$ et $(F, \leq_F)$, on peut construire un nouvel ordre partiel $(E \times F, \leq)$ appellé *ordre produit* défini par $(a, x) \leq (b, y) \equiv a \leq_E b \wedge x \leq_F y$.
On considère l'ordre $\{public, prive\}$ avec $public \leq prive$ et l'ordre naturel des parties de $\{compta, inge\}$ muni de l'inclusion $(\wp(\{compta, inge\}), \subseteq)$. Dessiner leurs diagrammes de Hasse respectifs ainsi que l'ordre produit obtenu.
5. Quel est l'intérêt de cette construction pour le contrôle d'accès à niveaux ?

4.2 Modèles à rôles

Exercice 41 : Hiérarchisation des rôles

On dispose d'un politique RBAC dont on donne les deux relations *URA* et *PRA* ci dessous. On note les permissions sous la forme *ao* où *a* est une action et *o* un objet.

	I	M	G	P	S
Alice	×	×			
Bob	×		×		
Charly	×			×	
Denise					×

	r1	w1	r2	w2	r3	w3	r4	w4
Infirmier	×		×		×			
Médecin		×						
Gastrologue				×			×	×
Pédiatre						×	×	×
Secrétaire					×		×	

1. Calculer la matrice des droits.
2. On tâche de réorganiser la politique RBAC pour intégrer la hiérarchisation des rôles. Proposer une hiérarchie compatible avec les matrices données où chaque utilisateur est directement affecté à un unique rôle.
3. Dans une hiérarchie de rôles, il est possible de définir des rôles auxquels aucun utilisateur n'est affecté directement ou auxquels aucun privilège n'est affecté directement. Commenter l'intérêt de ces types de rôles particuliers. Comment s'affranchir des rôles sans privilèges directs ?

Exercice 42 : Erreurs dans une politique ([2])

La politique suivante a été définie :

$$
\begin{aligned}
U =& \{Lee, Mike, Nell, Omar, Pat\} \\
P =& \{p_i \mid i \in [0..9]\} \\
R =& \{A, B, C, D, E, F, G, H, J\} \\
URA =& \{(Lee, D), (Lee, B), (Mike, F), (Nell, D), (Nell, F), (Mike, J), \\
& (Pat, H), (Pat, J), (Omar, A), (Omar, H)\} \\
PRA =& \{(p_1, E), (p_2, E), (p_3, D), (p_4, C), (p_5, G), (p_6, J), \\
& (p_7, A), (p_8, B), (p_9, F), (p_0, H)\} \\
Ssd_s =& \{(B, C), (G, A), (A, F), (G, F), (B, J)\} \\
\succeq_s =& \{(F, H), (B, F), (B, A), (C, A), (D, C), (D, G), (G, J), (E, B), (E, D)\}
\end{aligned}
$$

Ssd_s et $\succeq_s$ sont les squelettes respectifs de la relation d'exclusion mutuelle statique binaire *Ssd* entre rôles et de la relation d'héritage $\succeq$ entre rôles. Autrement dit, *Ssd* est la couverture symétrique de Ssd_s et $\succeq$ la couverture transitive et réflexive de $\succeq_s$.

Le système est actuellement en production et trois sessions s_1, s_2 et s_3 sont actives :

$$
\begin{aligned}
user(s_1) =& Omar \\
role(s_1) =& \{B, D\} \\
user(s_2) =& Nell \\
role(s_2) =& \{G, F\} \\
user(s_3) =& Pat \\
role(s_3) =& \{J, H\}
\end{aligned}
$$

2. http://www.cis.syr.edu/~sueo/774/

1. Dessiner le diagramme de Hasse de $\succeq$. Enrichir ce diagramme en ajoutant les exclusions mutuelles, les affectations d'utilisateurs et de permissions.
2. Identifier des erreurs dans l'état actuel du système.

Exercice 43 : Propriétés de l'exclusion mutuelle entre rôles ([GB98])

On considère les règles suivantes sur la relation d'exclusion mutuelle (statique) entre rôles :

1. aucun rôle n'est en exclusion avec lui même ;
2. la relation d'exclusion est symétrique ;
3. deux rôles en exclusion mutuelle n'héritent pas l'un de l'autre ;
4. aucun rôle n'hérite de deux rôles en exclusion mutuelle ;
5. l'exclusion est transmise par la relation d'héritage, c'est-à-dire que si un rôle s hérite de r_1 et que celui-ci est en exclusion avec r_2, alors s est aussi en exclusion avec r_2 ;

1. Formaliser ces règles en logique du premier ordre avec les prédicats $Ssd/2$ et $\succeq /2$ qui dénotent respectivement l'exclusion mutuelle et l'héritage entre rôles ($a \succeq b$ pour a hérite de b).
2. Montrer que les règles 3 et 4 peuvent être déduites des autres.
3. Quels intérêts peut-on trouver aux preuves de la question 2 ?

Exercice 44 : Modélisation avec les rôles : département informatique

Le département d'informatique a décidé d'investir dans un système de gestion électronique des notes. Voici les propriétés de contrôle d'accès que le département souhaite voir appliquées :

— à chaque unité d'enseignement U_i sont associés un ou plusieurs responsable(s) (U_iR) ainsi que un ou plusieurs chargé(s) de TD (U_iC) ;
— responsables et chargés de TD peuvent lire (RU_i) et écrire (WU_i) des notes pour cette UE mais seuls les responsables peuvent valider (SU_i) les notes ;
— tous les enseignants peuvent lire les notes, quelle que soit l'UE ;
— les étudiants en thèse peuvent être considéré comme des enseignants, toutefois, aucun étudiant en thèse ne peut être responsable d'une UE ;
— tous les enseignants peuvent être responsables et chargés de TD, sans limite du nombre d'UE. Ils peuvent être les deux à la fois.

On considère trois UEs U_1, U_2 et U_3, un ensemble R de rôles et P de permissions :

$$R = \{Ens, PhD\} \cup \{U_1R, U_1C, U_2R, U_2C, U_3R, U_3C\}$$
$$P = \{RU_1, WU_1, SU_1, RU_2, WU_2, SU_2, RU_3, WU_3, SU_3\}$$

Il s'agit de définir une politique RBAC qui répond au cahier des charges de ce système (ne pas faire d'hypothèse sur des droits supplémentaires).

1. Définir une hiérarchie de rôle ;
2. Définir une affectation de permissions aux rôles (PRA) ;
3. Définir une relation d'exclusion mutuelle statique binaire entre rôles ;
4. On souhaiterait ajouter les contraintes suivantes, comment les prendre en compte :
   — « les responsabilités d'UE sont incompatible »
   — « aucun étudiant en thèse ne peut être chargé de TD de plus d'une UE. »
   — « chaque UE doit avoir au moins un responsable. »

Exercice 45 : Modélisation avec les rôles : entreprise de plâtrerie & peintures

On considère la description de l'entreprise *Plâtrerie & Peintures Montiliennes* pour laquelle on va définir une politique RBAC dans une application de gestion.

- Alexandre (A) est un plâtrier qui est aussi quelques fois chef de chantier.
- Bernadette (B) est une électricienne.
- Charles (C) est un apprenti plombier qui est aussi quelques fois attaché de direction.
- Denis (D) est un plombier confirmé.
- Edouard (E) est un apprenti plâtrier et aussi un apprenti électricien.
- Françoise (F) est la chef de l'entreprise et quelques fois électricienne.

A cause de la prolifération de normes et règlements, un ensemble de règle interne définit quelles sont les opérations autorisées sur les chantiers. Les intitulés de permissions à utiliser par la suite sont donnés en *italique*.

- Les plâtriers peuvent poser des revêtement de *sol*, de *mur* et de *plafond*.
- Les électriciens peuvent installer des *câblages* ou des *disjoncteurs* et *tester* les câblages.
- Les plombiers peuvent poser de la *tuyauterie* et des *vannes*.
- Les apprentis peuvent *aider* et peuvent réaliser eux-mêmes quelques tâches :
  - les apprentis plâtriers peuvent poser des revêtements de `sol`,
  - les apprentis électriciens peuvent *tester* les câblages,
  - les apprentis plombiers peuvent poser de la *tuyauterie*.
- Les électriciens et les plombiers peuvent réaliser les *plans* de leurs installations.
- Les chefs de chantier peuvent *superviser* les travaux et *aider* quand nécessaire.
- Les attachés de direction peuvent réaliser les *plannings*, commander les *fournitures* et *facturer* les clients.
- Tous les employés peuvent *consulter* les plans.

1. Définir un ensemble de rôle hiérarchisés qui modélise ce problème. On donnera des noms évocateurs aux rôles et on précisera la hiérarchie en la dessinant. Définir les relations User-Role Assignment et Permission-Role Assignment en faisant en sorte que chaque permission soit associée à *exactement* un seul rôle. (*Note : pour respecter cette consigne il faut intégrer quelques rôles techniques qui ne correspondent pas directement à des métiers*)
2. On désire s'assurer d'une règle de séparation des tâches précisant que *plâtrier, électricien et plombier* sont des fonctions incompatibles. Discuter de la validité de cette règle et de sa réalisation sur le cas d'étude.

Exercice 46 : Implémentation de RBAC avec un SGBD-R

Pour une application de gestion, on souhaite implémenter un mécanisme de contrôle d'accès à rôles avec un SGBD Relationnel (SBGD-R). Les entités du modèle sont les utilisateurs, les sessions, les rôles et les permissions. On suppose que le SGBD dispose de *triggers* et de contraintes `CHECK`.

1. Proposer un modèle Entité/Association (E/A) ou UML d'un schéma de base de données pour stocker une politique RBAC sans hiérarchisation et sans exclusion
2. Donner les déclarations SQL correspondante en précisant les clefs primaires et étrangères.
3. Traduire formellement la contrainte d'intégrité "les utilisateurs ne peuvent endosser dans une session que des rôles qui leur sont attribués" et expliquer comment on peut la gérer.
4. Donner une requête SQL, une expression de l'algèbre relationnelle ou du calcul relationnel permettant de déterminer si un utilisateur `%u` a le droit exécuter l'action `%a` sur l'objet `%o` (on considérera ces paramètres comme des constantes dans les clauses `WHERE`).
5. Même question que précédemment mais lorsque les rôles sont hiérarchisés avec un nombre de niveaux hiérarchiques fixés (e.g., 3 niveaux dans RBAC Oracle). Préciser les modifications à apporter au schéma.

6. Quel problème est soulevé quand on souhaite gérer des hiérarchies de rôles sans limite du nombre de niveaux. Quelles solutions peut-on proposer pour résoudre ce problème ?

Exercice 47 : Modélisation des droits d'accès à la FST

On s'intéresse à une application centralisée de gestion des demandes de financement des travaux de recherche auprès de l'université Lyon 1. Vous participez à la réalisation de cette application, et en particulier du module de contrôle des accès.

L'organisation de l'université est réputée être une *structure* arborescente dont l'UBCL est la racine, les structures de niveau 2 sont les facultés et instituts qui lui sont immédiatement rattachées (exemples : Faculté des Sciences et Technologies – FST, Institut Universitaire Technologies – IUT, etc.) et les départements les structures de niveau 3 qui sont rattachées aux structures de niveau 2 (exemples pour la FST : Département Informatique, Département Physique, etc.).

Chaque structure (exemples : UCBL, IUT, FST, FST-Info, FST-Chimie, IUT-Biologie, etc.) est susceptible de financer des travaux de recherche. Le financement se fait sur examen d'un *dossier*. Chaque structure est dotée d'un unique *comité*, qui nomme en son sein des rapporteurs pour évaluer les dossiers. Après cette évaluation, les dossiers sont classés afin de déterminer lesquels seront financés.

Chaque *Enseignant-Chercheur* (E/C) est membre d'un département. Les E/C déposent des dossiers de demande de financement auprès des structures puis reçoivent la décision de financement quelques mois après. On ne peut déposer un dossier qu'à sa structure de rattachement ou aux structures parentes (exemple : un E/C FST-Info peut déposer auprès de FST-Info, de la FST ou de l'UCBL). Seuls peuvent être membres d'un comité les E/C des sous-structures de la structure du comité. Un E/C ne peut pas évaluer son propre projet.

Pour la modélisation, on considère les actions suivantes :

— `Deposer(idEC, numDossier, numComite)` quand un E/C dépose un dossier auprès d'un comité ;
— `AffecteC(idEC, numComite)` quand un E/C est nommé membre d'une commission ;
— `AffecteR(idEC, numDossier)` quand un E/C est nommé rapporteur d'un dossier déposé dans son comité ;
— `Rapporte(idEC, numDossier, note)` quand un E/C attribue une note à un dossier dont il est rapporteur.

1. Proposez un modèle relationnel de données pour stocker les *structures*, les *comités*, les *E/C*, les *dossiers* et les relations entres ces entités (exemples : les affectations des E/C aux départements, aux comités, etc.). Représentez graphiquement votre modèle en précisant le formalisme utilisé, les contraintes de clef primaire et étrangère. Un soin particulier sera attaché à la modélisation de la structure qui devra être justifié. On remarque notamment qu'il y a une bijection entre les *comités* et les *structures*.
2. Donner une requête SQL, ou à défaut un algorithme en pseudo-code, qui vérifie si un dépôt est autorisé, action `Deposer(idEC, numDossier, numComite)`.
3. Même question pour l'action `AffecteC(idEC, numComite)`.
4. Même question pour l'action `AffecteR(idEC, numDossier)`.
5. Même question pour l'action `Rapporte(idEC, numDossier, note)`.

4.3 Contrôle d'accès dans `web.xml`

Les descripteurs de déploiement `web.xml` des serveurs JEE contiennent une section *security constraints* destinée à contrôler les accès aux servlets que le serveur héberge. Ces servlets sont identifiées par leurs urls. La syntaxe abrégée des *security constraints* est donnée par la grammaire suivante :

```
<ac>   ::=  '*' | '<' <rl> '>'
<rl>   ::= <empty> | "role" ',' <rl>
<up>   ::=  <empty> | "part" | '*' | "part" '/' <up>
<upl>  ::= <up> | <up> ',' <upl>
<ml>   ::= "method" | "method" ',' <ml>
<wrc>  ::= '{' <upl> '}' '[' <empty> ']' | '{' <upl> '}' '[' <ml> ']'
<wrcl> ::= <wrc> | <wrc> ',' <wrcl>
<sc>   ::= <wrcl> | <wrcl> <ac>
<scl>  ::= <sc> | <sc> <scl>
```

On donne un exemple de telles contraintes.

```
POL = {SC1, SC2, SC3, SC4}
SC1 = {/*, /acme/wholesale/*, /acme/retail/*} [DELETE, PUT] <>
SC2 = {/acme/wholesale/*} [GET, PUT] <SALESCLERK>
SC3 = {/acme/wholesale/*} [GET, POST] <CONTRACTOR>
SC4 = {/acme/retail/*} [GET, POST] <CONTRACTOR, HOMEOWNER>
```

La sémantique informelle de ces contraintes est la suivante [CY03] :

> *An authorization constraint [...] names the authorization roles permitted to perform the constrained requests. A user must be a member of at least one of the named roles to be permitted to perform the constrained requests. The special role name "*" is a shorthand for all role names defined in the deployment descriptor. An authorization constraint that names no roles indicates that access to the constrained requests must not be permitted under any circumstances.*

Quand plusieurs motifs d'urls sont applicables, la règle de sélection est la suivante :

> *When a Servlet container receives a request, it shall use the algorithm described in SRV.11.1 to select the constraints (if any) defined on the url-pattern that is the best match to the request URI. If no constraints are selected, the container shall accept the request. Otherwise the container shall determine if the HTTP method of the request is constrained at the selected pattern. If it is not, the request shall be accepted. Otherwise, the request must satisfy the constraints that apply to the http-method at the url-pattern.*

Informellement, l'algorithme de la section *SRV.11.1* choisi la contrainte dont l'expression régulière est la plus spécifique possible (*most specific takes precedence*). Dans le cas où plusieurs règles sont applicables, la résolution est la suivante :

> *When a url-pattern and http-method pair occurs in multiple constraints, the constraints (on the pattern and method) are defined by combining the individual constraints. The rules for combining constraints in which the same pattern and method occur are as follows.*
>
> *The combination of authorization constraints that name roles or that imply roles via the name "*" shall yield the union of the role names in the individual constraints as permitted roles. A security constraint that does not contain an authorization constraint shall combine with authorization constraints that name or imply roles to allow unauthenticated access. The special case of an authorization constraint that names no roles shall combine with any other constraints to override their affects and cause access to be precluded.*

Exercice 48 : Questions

1. Pour faire le parallèle avec les modèles déjà vus, indiquer quels sont les sujets, actions et objets considérés dans ce modèle. Préciser l'ensemble des décisions.
2. Remplir le tableau suivant, en indiquant quels sont les rôles autorisés

url-pattern	http-method	roles
/*	DELETE	
/*	PUT	
/acme/wholesale/*	DELETE	
/acme/wholesale/*	GET	
/acme/wholesale/*	POST	
/acme/wholesale/*	PUT	
/acme/retail/*	DELETE	
/acme/retail/*	GET	
/acme/retail/*	POST	
/acme/retail/*	PUT	

3. On ajoute la règle suivante

   `SC5 = {/acme/} [GET] <HOMEOWNER>`

   Remplir le tableau suivant en conséquence :

url-pattern	http-method	roles
/acme/	PUT	
/acme/	DELETE	
/acme/	GET	

4. On désigne par $\mathcal{R}$ la collection des rôles existants et par $\mathcal{L}$ l'ensemble de tous les sous-ensembles de $\mathcal{R}$ auquel on ajoute un élément $\top$ pour les accès non authentifiés : $\mathcal{L} = \wp(\mathcal{R}) \cup \{\top\}$ avec $\top \notin \mathcal{R}$. Un ordre partiel $\leq$ sur $\mathcal{L}$ est défini par $R_A \leq R_B \equiv R_B = \top \vee (R_A \neq \top \wedge R_A \subseteq R_B)$. Définir (formellement ou en pseudo-code) l'opération $\otimes : \mathcal{L} \times \mathcal{L} \to \mathcal{L}$ qui combine deux éléments de $\mathcal{L}$ quand plusieurs règles sont applicables.
5. Sachant que l'exemple de la question 2 est à peu près le seul de la spécification, critiquer constructivement la sémantique des *security constraints* des descripteurs de déploiement.

4.4 Contrôle d'accès XACML

Exercice 49 : Combinaisons de politiques XACML

Une politique de contrôle d'accès XACML en syntaxe simplifiée est donnée en annexe. En XACML, une *PolicySet* (ensemble de politiques) est composée d'une *Target* (`Null` est une cible toujours évaluée à *True*), d'une collection ordonnée de *Policy* (politiques élémentaires) et d'un algorithme de combinaison de décision (*p-o* pour *permit-overrides* dans l'exemple).Une *Policy* est construite similairement à partir de *Rule* (règles).

Une *Rule* est composée d'une décision (`p` pour `Permit`, `d` pour `Deny`), d'une *Target* et d'une *Condition*. Une *Target* est une formule de logique des prédicats sans quantificateurs ni négation n'utilisant que `subject(S)`, `action(A)` et `resource(R)`. Une *Condition* est une formule de logique de prédicats arbitraires supposées évaluables avec l'interprétation usuelle (e.g., `age(Y) < 18`). Si une requête ne correspond pas à une *Target*, le résultat de l'évaluation est `NotApplicable`.

1. Soit la requête `{subject(doctor), action(write), resource(medical_record), doctor(id,d), patient(id,p), medical_record(id,p)}`. Donner le résultat de l'évaluation de cette demande d'accès sur `PS_patient` en le justifiant.
2. Identifier un cas où `P_patient_record` et `P_medical_record` sont applicables et donner la requête correspondante.

3. Soit $\mathcal{D} = \{\texttt{Permit}, \texttt{Deny}, \texttt{NotApplicable}\}$ l'ensemble des décisions et $V = [v_0, v_1, \ldots, v_n] \in \mathcal{D}^\star$ une liste ordonnée finie de décisions avec $v_i \in \mathcal{D}$. Donner l'algorithme `d-o` en pseudo code en style impératif.

4.5 Filtrage par pare-feux

Exercice 50 : Principes de la configuration (*[AJO10, ex. n° 33]*)

1. On considère les principes de base suivants, applicables notamment pour la configuration des pare-feux, les expliquer :
   1. moindre privilège ;
   2. interdiction par défaut ;
   3. défense en profondeur ;
   4. goulet d'étranglement ;
   5. simplicité ;
   6. participation des utilisateurs ;

Exercice 51 : Règles de filtrage (*[AJO10, ex. n° 35]*)

On considère un pare-feux sans mémoire qui abrite la machine `203.167.75.1`. L'utilisateur de cette machine souhaite naviguer sur le Web, recevoir et initier des connexions SSH de et vers Internet.

1. Donner la table de filtrage du pare-feu statique sous forme de tableau.
2. Pour chaque ligne, préciser si le pare-feu doit accepter les paquets dont le flag SYN est activé.

Chapitre 5

Protection de la vie privée

Exercice 52 : Analyse de décisions de la CNIL

L'objectif de cet exercice est d'une part de se familiariser avec la législation française sur les données personnelles, et d'autre part, d'être capable d'évaluer les risques sur la vie privée d'un système de traitement de données personnelles.

1. Pour chacune des trois décisions de la CNIL en annexe A.4, indiquer quelle est la décision ainsi que les extraits qui motivent la décision.
2. Evaluer chacune des décision d'après la grille de critères fournie en annexe A.5, identifier le(s) critère(s) les plus importants vis-à-vis de la décision.
3. Proposer quelques « slogans » à donner aux concepteurs de systèmes qui résumeraient les principaux critères de la grille.

5.1 Bases de données hippocratiques

En annexe est donné un exemple de schéma de base de données de ce que pourrait être une base de données hippocratique. Il ne s'agit pas d'une proposition finale mais d'un design préliminaire ayant vocation à soulever des problèmes et à créer la discussion.

Exercice 53 : Bases de données hippocratiques

1. Quelle est modification des schémas des données métier est proposée dans ce design ?
   La figure 5 est une politique de contrôle d'accès. On considère que les éléments du domaine de l'attribut *authorized-users* sont des rôles internes à l'entreprise qui utilise la BD.
2. Quels sont les objets de ce modèle de contrôle d'accès ? Quelles modifications à RBAC sont apportées avec ce design ?
3. Considérant une requête SQL, comment décider de l'autorisation ou du refus de l'exécution de la requête dans ce modèle ?
4. Comment assurer la limitation effective de la durée de rétention ?

Annexe A

Appendice

A.1 Traces `nmap`

Trace A

```
SENT (0.5190s) TCP 134.214.143.195:53 > 134.214.142.10:22 S ttl=64 id=45396 iplen=44
SENT (0.5190s) TCP 134.214.143.195:53 > 134.214.142.10:443 S ttl=64 id=23365 iplen=44
SENT (0.5190s) TCP 134.214.143.195:53 > 134.214.142.10:80 S ttl=64 id=53776 iplen=44
SENT (0.5190s) TCP 134.214.143.195:53 > 134.214.142.10:8080 S ttl=64 id=24307 iplen=44
SENT (0.5190s) TCP 134.214.143.195:53 > 134.214.142.10:631 S ttl=64 id=16206 iplen=44
RCVD (0.5190s) TCP 134.214.142.10:22 > 134.214.143.195:53 SA ttl=64 id=0 iplen=44
RCVD (0.5190s) TCP 134.214.142.10:443 > 134.214.143.195:53 SA ttl=64 id=0 iplen=44
RCVD (0.5190s) TCP 134.214.142.10:80 > 134.214.143.195:53 SA ttl=64 id=0 iplen=44
RCVD (0.5190s) TCP 134.214.142.10:8080 > 134.214.143.195:53 SA ttl=64 id=0 iplen=44
RCVD (0.5200s) TCP 134.214.142.10:631 > 134.214.143.195:53 RA ttl=64 id=0 iplen=40
```

Trace B

```
SENT (0.2300s) TCP 134.214.235.21:53 > 134.214.142.10:443 S ttl=64 id=49208 iplen=44
SENT (0.2300s) TCP 134.214.235.21:53 > 134.214.142.10:80 S ttl=64 id=32073 iplen=44
SENT (0.2300s) TCP 134.214.235.21:53 > 134.214.142.10:8080 S ttl=64 id=30178 iplen=44
SENT (0.2300s) TCP 134.214.235.21:53 > 134.214.142.10:22 S ttl=64 id=38225 iplen=44
SENT (0.2300s) TCP 134.214.235.21:53 > 134.214.142.10:631 S ttl=64 id=9262 iplen=44
RCVD (0.2320s) TCP 134.214.142.10:443 > 134.214.235.21:53 SA ttl=64 id=0 iplen=44
RCVD (0.2330s) TCP 134.214.142.10:80 > 134.214.235.21:53 SA ttl=64 id=0 iplen=44
RCVD (0.2340s) TCP 134.214.142.10:8080 > 134.214.235.21:53 SA ttl=64 id=0 iplen=44
SENT (1.3310s) TCP 134.214.235.21:53 > 134.214.142.10:631 S ttl=64 id=39325 iplen=44
SENT (1.3310s) TCP 134.214.235.21:53 > 134.214.142.10:22 S ttl=64 id=370 iplen=44
```

A.2 Syntaxe concrète `web.xml`

```
<security-constraint>
  <web-resource-collection>
    <web-resource-name>restricted methods</web-resource-name>
    <url-pattern>/*</url-pattern>
    <url-pattern>/acme/wholesale/*</url-pattern>
    <url-pattern>/acme/retail/*</url-pattern>
    <http-method>DELETE</http-method>
    <http-method>PUT</http-method>
  </web-resource-collection>
  <auth-constraint/>
</security-constraint>
<security-constraint>
```

```
  <web-resource-collection>
    <web-resource-name>wholesale</web-resource-name>
    <url-pattern>/acme/wholesale/*</url-pattern>
    <http-method>GET</http-method>
    <http-method>PUT</http-method>
  </web-resource-collection>
  <auth-constraint>
    <role-name>SALESCLERK</role-name>
  </auth-constraint>
</security-constraint>
<security-constraint>
  <web-resource-collection>
    <web-resource-name>wholesale</web-resource-name>
    <url-pattern>/acme/wholesale/*</url-pattern>
    <http-method>GET</http-method>
    <http-method>POST</http-method>
  </web-resource-collection>
  <auth-constraint>
    <role-name>CONTRACTOR</role-name>
  </auth-constraint>
</security-constraint>
<security-constraint>
  <web-resource-collection>
    <web-resource-name>retail</web-resource-name>
    <url-pattern>/acme/retail/*</url-pattern>
    <http-method>GET</http-method>
    <http-method>POST</http-method>
  </web-resource-collection>
  <auth-constraint>
    <role-name>CONTRACTOR</role-name>
    <role-name>HOMEOWNER</role-name>
  </auth-constraint>
</security-constraint>
```

A.3 Exemple XACML

```
PS_patient = <Null, <P_patient_record, P_medical_record>, p-o>
P_patient_record = <Null, <RP1, RP2, RP3>, d-o>
P_medical_record = <Null, <RM1, RM2>, d-o>

RP1 =
  < p,
  subject(patient) /\ action(read) /\ resource(patient_record),
  patient(id,X) /\ patient_record(id,Y) /\
  (X = Y \/ (age(Y) < 18 /\ guardian(X,Y))>

RP2 =
  < p,
  subject(patient) /\ action(write) /\ resource(patient_survey),
  patient(id,X) /\ patient_survey(id, X)>

RP3=
  < p,
```

```
  (subject(doctor) \/ subject(nurse)) /\ action(read) /\ resource(patient_record),
  true>

RM1 =
  < p,
  subject(doctor) /\ action(write) /\ resource(medical_record),
  doctor(id,X) /\ patient(id,Y) /\ medical_record(id, Y) /\ patient_doctor(Y,X)>

RM2 =
  < d,
  subject(doctor) /\ action(write) /\ resource(medical_record),
  doctor(id,X), patient(id,Y), medical_record(id, Y), not
  patient_doctor(Y,X)>
```

A.4 Délibérations de la CNIL

Commission Nationale de l'Informatique et des Libertés

Délibération n°2007-091 du 25 avril 2007

Délibération refusant la mise en oeuvre par l'Autorité de contrôle des assurances et des mutuelles CAM) d'un traitement automatisé de données à caractère personnel reposant sur la reconnaissance des empreintes digitales et ayant pour finalité le contrôle de l'accès au réseau informatique (postes de travail fixes et portables).

La Commission nationale de l'informatique et des libertés,

Vu la Convention n° 108 du Conseil de l'Europe pour la protection des personnes à l'égard du traitement automatisé des données à caractère personnel ;

Vu la directive 95/46/CE du Parlement européen et du Conseil du 24 octobre 1995 relative à la protection des personnes physiques à l'égard du traitement de données à caractère personnel et la libre circulation de ces données ;

Vu la loi n° 78-17 du 6 janvier 1978 relative à l'informatique, aux fichiers et aux libertés modifiée par la loi n° 2004-801 du 6 août 2004 relative à la protection des personnes physiques à l'égard des traitements de données à caractère personnel, notamment son article 25-8° ;

Vu le décret n° 2005-1309 du 20 octobre 2005 pris pour l'application de la loi n° 78-17 du 6 janvier 1978 relative à l'informatique, aux fichiers et aux libertés, modifiée par la loi n° 2004-801 du 6 août 2004 modifié par le décret n° 2007-451 du 25 mars 2007 ;

Vu la demande d'autorisation, présentée par l'Autorité de contrôle des assurances et des mutuelles (ACAM) d'un traitement automatisé de données à caractère personnel reposant sur la reconnaissance des empreintes digitales et ayant pour finalité le contrôle de l'accès au réseau informatique (postes de travail fixes et portables) ;

Après avoir entendu M. Hubert BOUCHET, commissaire en son rapport et Mme Pascale COMPAGNIE, commissaire du Gouvernement, en ses observations.

Formule les observations suivantes :

La Commission nationale de l'informatique et des libertés a été saisie par l'Autorité de contrôle des assurances et des mutuelles (ACAM) d'un traitement automatisé de données à caractère personnel reposant sur la reconnaissance des empreintes digitales et ayant pour finalité le contrôle de l'accès au réseau informatique (postes de travail fixes et portables).

Il y a lieu de faire application des dispositions prévues à l'article 25-8° de la loi du 6 janvier 1978 modifiée qui soumet à autorisation les traitements comportant des données biométriques nécessaires au contrôle de l'identité des personnes.

L'ACAM est chargée de contrôler le respect par les entreprises d'assurance et de réassurance, les mutuelles, les institutions de prévoyance et les institutions de prévoyance et les institutions de retraite supplémentaire, des dispositions législatives et réglementaires qui leur sont applicables et de sanctionner les manquements constatés.

Le dispositif a pour objet de contrôler l'accès au réseau informatique (postes de travail fixes et portables).

La Commission considère que la constitution de bases de données d'empreintes digitales ne peut être admise que dans certaines circonstances particulières où l'exigence d'identification des personnes résulte d'un fort impératif de sécurité, conformément aux dispositions de l'article 6-3° de la loi du 6 janvier 1978 modifiée. En effet, cet article dispose que les traitements ne peuvent porter que sur des données à caractère personnel adéquates, pertinentes et non excessives au regard des finalités pour lesquelles elles sont collectées et de leurs traitements ultérieurs.

En l'espèce, l'objectif poursuivi par l'ACAM tendant au contrôle de l'accès au réseau informatique (postes de travail fixes et portables), s'il est légitime, n'est associé à aucune circonstance particulière justifiant la conservation dans une base de données des empreintes digitales des employés habilités à accéder au réseau informatique (postes de travail fixes et portables). En conséquence, le traitement pris dans son ensemble n'apparaît ni adapté ni proportionné à l'objectif poursuivi.

Dès lors, la Commission n'autorise pas, en l'état, l'Autorité de contrôle des assurances et des mutuelles (ACAM) sise au 54 rue de Châteaudun 75436 Paris Cedex 09, à mettre en oeuvre un traitement de données à caractère personnel reposant sur l'utilisation d'un dispositif biométrique de reconnaissance des empreintes digitales et dont la finalité est le contrôle de l'accès au réseau informatique (postes de travail fixes et portables).

Le Président, Alex TURK.

Nature de la délibération: Refus d'autorisation

Commission Nationale de l'Informatique et des Libertés

Délibération n°2005-169 du 05 juillet 2005

Délibération portant autorisation de mise en oeuvre par le collège "Les Mimosas" d'un traitement automatisé de données à caractère personnel reposant sur la reconnaissance du contour de la main et ayant pour finalité de contrôler l'accès au restaurant scolaire.

La Commission nationale de l'informatique et des libertés,

Vu la convention n° 108 du Conseil de l'Europe pour la protection des personnes à l'égard du traitement automatisé des données à caractère personnel ;

Vu la directive 95/46/CE du Parlement européen et du Conseil du 24 octobre 1995 relative à la protection des personnes physiques à l'égard du traitement de données à caractère personnel et à la libre circulation de ces données ;

Vu la loi n° 78-17 du 6 janvier 1978 relative à l'informatique, aux fichiers et aux libertés modifiée par la loi n° 2004-801 du 6 août 2004 relative à la protection des personnes physiques à l'égard des traitements de données à caractère personnel, et notamment son article 25-8° ;

Vu la déclaration présentée par le principal du collège "Les Mimosas", d'un traitement automatisé de données à caractère personnel ayant pour finalité le contrôle de l'accès au restaurant scolaire par la reconnaissance du contour de la main ;

Après avoir entendu M. Francis Delattre, Commissaire, en son rapport, et Mme Charlotte-Marie Pitrat, Commissaire du Gouvernement, en ses observations,

Formule les observations suivantes :

Le collège "Les Mimosas", situé à Mandelieu, a saisi la CNIL d'une déclaration relative à la mise en oeuvre d'un traitement de données à caractère personnel ayant pour finalité le contrôle de l'accès des élèves et des personnels au restaurant scolaire par le recours à un dispositif biométrique reposant sur la reconnaissance du contour de la main.

La Commission considère qu'il y a lieu de faire application des dispositions prévues à l'article 25-8° de la loi du 6 janvier 1978 modifiée qui soumet à autorisation les traitements comportant des données biométriques nécessaires au contrôle de l'identité des personnes.

Le système envisagé reposera d'une part, sur la mise en oeuvre d'un fichier de gestion recensant les élèves et les personnels fréquentant la cantine scolaire et d'autre part, sur un dispositif de contrôle d'accès. Ce dernier sera composé d'une borne d'accès, située à l'entrée du restaurant, reliée à un lecteur biométrique, lequel contiendra une base de données comportant les gabarits biométriques et les codes d'accès. L'enregistrement de l'image de la main de chaque personne sera effectué au début de l'année scolaire.

Le contour de la main fait partie des données qui ne laissent pas de traces susceptibles d'être utilisées à des fins étrangères à la finalité recherchée par le responsable du traitement. Le lecteur biométrique utilisé sera accompagné d'un dispositif de nature à garantir la sécurité des données.

Le recours à la technique de reconnaissance du contour de la main permettra de s'assurer que les données nécessaires au contrôle de l'accès ne sont ni perdues, ni échangées et que seules les personnes habilitées peuvent accéder au service.

Les personnes concernées qui ne seront pas désireuses d'utiliser la technologie biométrique seront dotées d'une carte à code-barre pour accéder au restaurant scolaire.

Compte tenu des caractéristiques du dispositif présenté à la Commission et en l'état actuel des connaissances sur la technologie utilisée, la mise en oeuvre d'un traitement reposant sur la reconnaissance de la géométrie de la main est adaptée et proportionnée à la finalité assignée au dispositif.

Les droits d'accès et de rectification s'exerceront auprès du principal du collège.

Les catégories de données à caractère personnel enregistrées seront,

- s'agissant des élèves : les gabarits biométriques de la main associés à un code d'accès personnel, et les données de gestion utiles pour l'accès au restaurant, à savoir, l'identité de l'élève, la classe, le numéro d'ordre dans l'établissement, les coordonnées du responsable légal, un code horaire et un code tarif.

S'agissant des personnels : les gabarits biométriques de la main associés à un code d'accès personnel, l'identité, les codes horaire et tarif.

Les destinataires des informations seront les seuls utilisateurs du dispositif autorisés : le principal et le conseiller principal d'éducation.

Autorise, dans ces conditions, le collège "Les Mimosas", à mettre en oeuvre un traitement de données à caractère personnel ayant pour finalité de contrôler l'accès au restaurant scolaire par la mise en oeuvre d'un dispositif de reconnaissance du contour de la main.

Le président, Alex TURK.

Nature de la délibération: Autorisation

Commission Nationale de l'Informatique et des Libertés

Délibération n°2005-111 du 26 mai 2005

Délibération relative à une demande d'autorisation de la Compagnie européenne d'accumulateurs pour la mise en oeuvre d'un dispositif de "ligne éthique"

La Commission nationale de l'informatique et des libertés,

Saisie le 29 juillet 2004 d'une déclaration portant sur la mise en oeuvre d'un dispositif de "ligne éthique" au sein de la Compagnie européenne d'accumulateurs,

Vu la Convention n° 108 du Conseil de l'Europe pour la protection des personnes à l'égard du traitement automatisé des données à caractère personnel,

Vu la directive 95/46/CE du Parlement européen et du Conseil du 24 octobre 1995 relative à la protection des personnes physiques à l'égard du traitement de données à caractère personnel et la libre circulation de ces données,

Vu la loi n° 78-17 du 6 janvier 1978 relative à l'informatique, aux fichiers et aux libertés modifiée par la loi n° 2004-801 du 6 août 2004 relative à la protection des personnes physiques à l'égard des traitements de données à caractère personnel,

Après avoir entendu M. Hubert Bouchet, commissaire, en son rapport, et Mme Charlotte-Marie Pitrat, commissaire du Gouvernement, en ses observations,

Formule les observations suivantes :

Sur le dispositif présenté

La Compagnie européenne d'accumulateurs (CEAC) a saisi la CNIL d'une déclaration concernant la mise en oeuvre d'une "hotline" (ligne téléphonique dédiée) à destination de ses 1500 employés.

Ce dispositif de "ligne éthique", conçu par sa société mère Exide Technologies afin de se conformer aux dispositions de la loi américaine dite "Sarbanes-Oxley", devrait permettre à l'ensemble des salariés du groupe "de communiquer avec le comité de surveillance comptable du conseil d'administration d'Exide sur des sujets tels que les inexactitudes ou les irrégularités comptables qui pourraient être commises".

La "hotline" devrait également permettre aux salariés d'alerter les dirigeants du groupe sur les éventuelles violations des principes en vigueur dans l'entreprise (règles de conduite éthique ou commerciale) ou des lois en vigueur.

Le dispositif s'appuierait à la fois sur un numéro vert et sur une adresse électronique.

Dans les deux cas, les alertes et les demandes d'information seraient en fait adressées à un sous-traitant américain pour le compte de Exide Technologies. S'agissant des appels passés en langue française, un second prestataire de service américain interviendrait.

S'il le souhaitait, l'anonymat de l'appelant serait garanti.

Les sous-traitants seraient chargés d'enregistrer sur support informatique le contenu des demandes et des alertes selon la classification suivante : "(1) ressources humaines ou problèmes de travail, (2) fraude ou vol, (3) erreur comptable, (4) problèmes liés aux principes de conduite et d'éthique".

En fonction de cette classification, un résumé écrit des appels et des messages électroniques reçus devrait ensuite être transmis, par e-mail crypté, aux personnes nommément désignées à cet effet par la société-mère (département juridique, département comptabilité, comité international, comité de vérification des comptes du conseil d'administration).

Le destinataire de l'information au sein d'Exide Technologies réaliserait ensuite, le cas échéant, une enquête interne. Celle-ci s'effectuerait en liaison avec le responsable juridique France (CEAC) qui recevrait les données nécessaires par courrier électronique.

Un "suivi de dossier" serait également adressé, par voie électronique, par la société-mère au responsable juridique France, qui le transmettrait au responsable des ressources humaines France.

Tout salarié concerné par un appel serait informé "le plus tôt possible des allégations prononcées à son encontre de telle sorte qu'il puisse s'expliquer".

Enfin, la durée de conservation des données serait limitée à une année.

Sur la détermination du responsable de traitement et l'application de la loi du 6 janvier 1978

L'article 3 de la loi du 6 janvier 1978 modifiée dispose que le responsable d'un traitement de données à caractère personnel est, sauf désignation expresse par les dispositions législatives ou réglementaires relatives à ce traitement, la personne, l'autorité publique, le service ou l'organisme qui détermine ses finalités et ses moyens.

Il ressort du dossier de formalités préalables présenté par la société CEAC que cette dernière agit auprès de la CNIL en qualité de responsable du dispositif de "ligne éthique" qu'elle envisage de mettre en oeuvre, et en particulier des traitements de données opérés lors des enquêtes diligentées sur des employés déterminés à la suite d'un signalement opéré dans le cadre du dispositif.

Dès lors, la Commission constate que la loi du 6 janvier 1978 relative à l'informatique, aux fichiers et aux libertés est applicable au dispositif de "ligne éthique" présenté et qu'elle est donc compétente pour se prononcer sur la conformité du projet à cette loi.

Sur la procédure déclarative applicable

La Commission relève que le dispositif envisagé est susceptible de conduire la société CEAC à décider, au titre des mesures correctives qu'elle doit prendre à la suite d'un signalement, à exclure des employés considérés fautifs du bénéfice de leur contrat de travail en l'absence de toute disposition législative ou réglementaire encadrant ce type de traitement.

Dès lors, la procédure d'autorisation prévue à l'article 25-1, 4° de la loi du 6 janvier 1978 modifiée doit être appliquée au traitement de données personnelles présenté.

Sur la conformité du dispositif présenté à la loi du 6 janvier 1978

La Commission considère que la mise en oeuvre par un employeur d'un dispositif destiné à organiser auprès de ses employés le recueil, quelle qu'en soit la forme, de données personnelles concernant des faits contraires aux règles de l'entreprise ou à la loi imputables à leurs collègues de travail, en ce qu'il pourrait conduire à un système organisé de délation professionnelle, ne peut qu'appeler de sa part une réserve de principe au regard de la loi du 6 janvier 1978 modifiée, et en particulier de son article 1er.

En ce sens, la Commission observe que la possibilité de réaliser une "alerte éthique" de façon anonyme ne pourrait que renforcer le risque de dénonciation calomnieuse.

Au surplus, la Commission estime que le dispositif présenté est disproportionné au regard des objectifs poursuivis et des risques de dénonciations calomnieuses et de stigmatisation des employés objets d'une "alerte éthique". Elle relève à cet égard que d'autres moyens prévus par la loi existent d'ores et déjà afin de garantir le respect des dispositions légales et des règles fixées par l'entreprise (actions de sensibilisation par l'information et la formation des personnels, rôle d'audit et d'alerte des commissaires aux comptes en matière financière et comptable, saisine de l'inspection du travail ou des juridictions compétentes).

La Commission relève enfin que les employés objets d'un signalement ne seraient, par définition, pas informés dès l'enregistrement de données mettant en cause leur intégrité professionnelle ou de citoyen, et n'auraient donc pas les moyens de s'opposer à ce traitement de données les concernant. Les modalités de collecte et de traitement de ces données, dont certaines pourraient concerner des faits susceptibles d'être constitutifs d'infractions pénales, ne peuvent dès lors être considérées comme loyales au sens de l'article 6 de la loi du 6 janvier 1978 modifiée.

Compte tenu de ces observations, la Commission n'autorise pas la mise en oeuvre du dispositif de "ligne éthique" présenté par la Compagnie européenne d'accumulateurs.

Le président, Alex TURK.

Nature de la délibération: Refus d'autorisation

A.5 Grille de critères pour l'analyse de la vie privée

ANALYSE DE RISQUE

0 : absence de risque/bénéfice important, 1 : risque potentiel/bénéfice commun, 2 : risque majeur/aucun bénéfice	
Domaines	
Abstract	
Résumé	

CRITÈRE / EVALUATION	2	1	0	PRÉCISION DU CRITÈRE	OBS°
1. RISQUES LIÉS AUX DONNÉES					
1.1 Sensibilité au sens de la loi				Données visées aux articles 8 et 9 (L. 78-17)	
1.2 Risques intrinsèques pour le sujet				Autres données sensibles : image, données biométriques, génétiques, bancaires, image, NIR	
1.3 « Linkability »				Capacité à lier les données avec d'autres	
2. RISQUES LIÉS À L'UTILISATION DES DONNÉES					
2.1 Risques liés aux finalités				Exclusion du bénéfice d'un droit ou d'une prestation	
2.2 Bénéfices pour la personne concernée				Le traitement présente un intérêt majeur, commun ou est-il sans intérêt pour la personne concernée ?	
2.3 Bénéfices pour le responsable				Le traitement présente un intérêt majeur, commun ou est-il sans intérêt pour le responsable ?	
2.4 Bénéfices pour la collectivité				Le traitement sert-il l'intérêt général ?	
3. DEGRÉ DE CONTRÔLE PAR LE SUJET					
3.1 Information et droit d' accès				L'information est-elle claire, complète et précise ? Le sujet peut-il exercer ses droits d'accès et de rectification ?	
3.2 Rapport de force avec le sujet				Traitement obligatoire, accédant en position de force très nette ; accédant en position de force	
3.3 Support local ou distant				La personne a-telle la maîtrise du support ou le traitement est-il centralisé par le responsable ?	
4. RISQUES LIÉS AU CONTEXTE TECHNIQUE					
4.1 Sécurité du système informatique				Pas de preuve de sécurisation, système sécurisé (label), système très sécurisé (Ex. certificat critères communs de sécurité EAL4+)	
4.2 Durée de conservation				Durée < à 2 jours ; < à 1 mois ou > à un mois	
5. RISQUES LIÉS AU CONTEXTE SOCIO-ÉCONOMIQUE					
5.1 Valeur commerciale des données				Forte valeur (profil de consommation), valeur commune (quartier, résidence, profession), faible valeur (sexe...)	
5.2 Degré de fiabilité des accédants				Accédant classé à risque, sans brevet de fiabilité ou bénéficiant d'antécédents favorables	
5.3 Collusion potentielle entre accédants				Capacité des accédants à nouer des relations avec avec d'acteurs susceptibles de détenir des données (forte, commune, faible). Risque de détournement, transfert hors UE	

Bibliographie

[Age10a] Agence Nationale de la Sécurité des Systèmes d'Information. Expression des Besoins et Identification des Objectifs de Sécurité – Bases De Connaissances. Technical report, 2010.

[Age10b] Agence Nationale de la Sécurité des Systèmes d'Information. Expression des Besoins et Identification des Objectifs de Sécurité – Étude de cas Archimed. Technical report, 2010.

[Age10c] Agence Nationale de la Sécurité des Systèmes d'Information. Expression des Besoins et Identification des Objectifs de Sécurité – Méthode de Gestion des Risques. Technical report, 2010.

[AJO10] Gildas Avoine, Pascal Junod, and Philippe Balbiani Oechslin. *Sécurité informatique*. Vuibert, deuxième edition, 2010.

[CY03] Danny Coward and Yutaka Yoshida. Java servlet specification, version 2.4. Technical report, Sun Microsystems, Inc, November 2003.

[DH76] Whitfield Diffie and Martin E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22(6) :644–654, 1976.

[GB98] Serban I. Gavrila and John F. Barkley. Formal specification for role based access control user/role and role/role relationship management. In *RBAC'98 : 3rd ACM workshop on Role-based access control*, pages 81–90, 1998.

[GH11] Solange Ghernaouti-Hélie. *Sécurité informatique et réseaux*. Dunod, troisième edition, 2011.