M2-TIW4 sécurité des systèmes d'informations
Contrôle continu final – durée 1h30

Master Technologie de l'Information, promotion 2018 – 2019

Mercredi 13 mars 2019

Ne rendez pas le sujet avec la copie.
Les documents ne sont pas autorisés sauf les dictionnaires de traduction. La composition en anglais est autorisée. Le barème sur 40 est indicatif. Il y a jusqu'à 4 points bonus.

1 Cryptographie

On rappelle que RSA est un système de chiffrement à clef publique dont le 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 $\mathcal{E}(m) = m^e \pmod{n}$ et celle de déchiffrement comme $\mathcal{D}(c) = c^d \pmod{n}$. On cite ensuite cet extrait de Wikipedia de l'article sur le *chiffrement homomorphe*.

> *En cryptographie, un* chiffrement homomorphe *est un chiffrement qui possède certaines caractéristiques algébriques qui le font commuter avec une opération mathématique, c'est-à-dire que le déchiffrement du résultat de cette opération sur des données chiffrées donne le même résultat que cette opération sur les données non chiffrées.*

Exercice 1 : Généralités (/6)

1. (/2) Pour chacune des réponses suivantes *justifier sur la copie* si elle est vrai ou fausse en expliquant la propriété en question. « Un protocole cryptographique peut permettre d'assurer ... »
   - ☐ L'intégrité des communications ;
   - ☐ L'authenticité des communications ;
   - ☐ La confidentialité des communications ;
   - ☐ La non-répudiation des communications ;

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. (/2) Pour chacune des réponses suivantes *justifier sur la copie* si elle est vrai ou fausse. « Alice utilise un chiffrement asymétrique, elle perd sa clef privée ... »
   - ☐ Alice peut toujours chiffrer des courriers qu'elle envoie ;
   - ☐ Alice peut toujours déchiffrer des courriers qu'elle reçoit ;
   - ☐ Alice peut toujours signer des courriers qu'elle envoie ;
   - ☐ Alice peut toujours vérifier la signature des courriers qu'elle reçoit ;
3. (/2) Expliquer pourquoi les protocoles comme PGP mêlent cryptographie à clef secrète et à clef publique.

Exercice 2 : Chiffrement (homomorphe) RSA (/12 + 2)

1. (/1) Expliquer pourquoi les fonctions sont $\mathcal{E}(m)$ notées et $\mathcal{D}(m)$, c'est-à-dire justifier le choix des symboles $\mathcal{E}$, m, $\mathcal{D}$ et c.
2. (/2) On prouve, en utilisant quelques théorèmes classiques d'arithmétique, que $(\mathcal{D}\circ\mathcal{E})(x) = x$. Qu'a-t-on prouvé ? Est-ce important de prouver que $(\mathcal{E} \circ \mathcal{D})(x) = x$?
3. (/2) Le problème de trouver x tel que $y = x^{\alpha} \pmod{p}$ (pour α et p donnés) est appelé le problème de *l'extraction des racines modulaires*. Expliquer en quoi la sécurité de RSA repose sur ce problème.
4. (/2) Quel est l'intérêt du chiffrement homomorphe ? Donner un exemple d'application dans le cloud.
5. (/2 + 2) Le chiffrement RSA tel que présenté est homomorphe. Expliquer en indiquant quelle est l'opération mathématique qui commute. Bonus (/2) : le prouver.
6. (/3) Vous devez implémenter RSA sur un dispositif embarqué pour chiffrer les données stockées sur une carte SD. Vous tâchez d'abord de convaincre votre autorité hiérarchique que ce n'est pas une bonne idée. Donnez les *trois arguments* qui vous paraissent les plus importants.

2 Contrôle d'accès

Pour formaliser les modèles à rôles avec hiérarchie (RBAC), on définit les ensembles U les utilisateurs, P des permissions, R des rôles et leur hiérarchie $\succeq \subseteq R \times R$ qui constitue un ordre partiel sur R ainsi que les relations $Pra \subseteq R \times P$ et $Ura \subseteq U \times R$. Concernant l'ordre $\succeq$, on peut lire $r_1 \succeq r_2$ comme « r_1 *est plus privilégié que* r_2 ». On définit également les applications $R \to 2^U$ et $R \to 2^P$ suivantes :

$$\begin{aligned}
\text{assigned_users}(r) &= \{u \in U \mid (u, r) \in Ura\} \\
\text{assigned_permissions}(r) &= \{p \in P \mid (r, p) \in Pra\} \\
\text{auth_users}(r) &= \{u \in U \mid \exists r'.r' \succeq r \wedge (u, r') \in Ura\} \\
\text{auth_permissions}(r) &= \{p \in P \mid \exists r'.r \succeq r' \wedge (r', p) \in Pra\}
\end{aligned}$$

Si pour RBAC le niveau de granularité de P est satisfaisant, pour comparer les modèles, on pourra définir deux ensembles additionnels A des actions et O des permissions et définir que $P = A \times O$, autrement dit chaque permission $p = (a, o) \in P$ est un couple formé d'une action $a \in A$ et d'un objet $o \in O$. On rappelle que la définition d'un modèle MAC à niveaux est formalisée par un ensemble L de niveaux, d'un ordre partiel $\sqsupseteq \subseteq L \times L$ entre ces niveaux ainsi que par deux fonctions d'attribution $L^U : U \to L$ et $L^O : O \to L$.

Exercice 3 : Généralités (/7)

1. (/2) Décrire ce qu'est un *moniteur de référence* en contrôle d'accès. Donner et expliquer les principales propriétés qu'il doit respecter.

2. (/2) On considère un modèle de type MAC à niveau, avec la hiérarchie *Très Secret Défense* (TS), *Secret Défense* (S), *Confidentiel Défense* (C) et *Non-classé* (U), on suppose que l'on a deux utilisateurs u_0 (accrédité S) et u_1 (accrédité C) ainsi que deux fichiers f_0 (de niveau C) et f_1 (de niveau U). Pour chacune des réponses suivantes *justifier sur la copie* si elle est vrai ou fausse.
   - ☐ u_1 peut lire et écrire dans f_0 ;
   - ☐ u_0 peut écrire dans f_1 ;
   - ☐ tous ceux qui peuvent lire f_0 peuvent aussi lire f_1 ;
   - ☐ u_0 peut lire tout ce que u_1 peut écrire ;
3. (/2) On considère maintenant une politique RBAC $\mathcal{R}$ avec les affectations suivantes $Ura = \{(u_0, r_0), (u_1, r_1), (u_2, r_2), (u_2, r_1)\}$ et la hiérarchie de rôles $\succeq = \{(r, r) \mid r \in R\} \cup \{(r_1, r_0), (r_2, r_0)\}$. Représenter graphiquement cette affectation *en expliquant le formalisme* employé.
4. (/1) Calculer auth_users(r_0).

Exercice 4 : Problème de modélisation (/15)

Dans cet exercice, on aura modélisé le modèle RBAC dans une base de données relationnelle avec les tables correspondant aux entités U, P, R, A et O (avec des attributs complémentaires associés) ainsi que les relations *Ura* et *Pra*. On aura par exemple pour U une table `User(user_id, first, last, ...)`. La table pour P sera de la forme `Perm(perm_id, id_action, id_object, ...)`. Concernant la hiérarchie de rôle $\succeq$ sur R, on la représente par une table `Senior(r_tgt, r_src)` et similairement pour l'ordre sur L défini par $\sqsupseteq$ qui est représenté par une table `Order(l_tgt, l_src)`.

Hypothèse simplificatrice : on suppose que les tables `Senior` et `Order` sont closes par transitivité[2] *et réflexivité*[3].

1. (/1) Dans la table `Perm` les attributs `id_action` et `id_object` sont des clefs étrangères, de plus le couple formé par ces attributs constitue aussi une clef. Justifier.
2. (/2) Expliquer pourquoi on peut représenter les affectations L^U et L^O avec des attributs supplémentaires aux tables représentant U et O. Préciser en particulier quelles sont les contraintes (relationnelles) qui doivent être respectées.
3. (/2) Donner la requête SQL qui correspond au calcul de auth_users et expliquer pourquoi l'hypothèse simplificatrice est bien simplificatrice.
4. (/2) Écrire une requête SQL qui permet de vérifier si la hiérarchie $\succeq$ représentée par `Senior` *est bien un arbre* (dont la racine est l'utilisateur le moins privilégié).
5. (/2) Si la relation $\succeq$ sur R est statique dans l'application (l'ensemble des rôles et sa hiérarchie sont fixés une bonne fois pour toutes, jamais modifiés), expliquer comment se passer de la table `Senior` en conservant les mêmes droits.
6. (/2) Donner la requête SQL qui calculer les triples d'autorisation $(u, a, o) \in U \times A \times O$ qui expriment qui a droit à quoi selon la politique RBAC.
7. (/2) Similairement, donner la requête SQL qui calcule les triples d'autorisation $(u, a, o) \in U \times A \times O$ qui expriment qui a droit à quoi selon la politique MAC à niveaux.
8. (/2) Les deux requêtes précédentes sont transformées en vues. Expliquer comment implémenter avec SQL un moniteur qui autorise l'accès d'un utilisateur à un objet *si les* deux *modèles RBAC et MAC à niveaux l'autorisent*. Expliquer comment réaliser la stratégie *si RBAC l'autorise ou si RBAC ne l'autorise pas mais que MAC à niveaux l'autorise*.

2. $\forall r_1.\forall r_2.\forall r_3.r_1 \succeq r_2 \wedge r_2 \succeq r_3 \Rightarrow r_1 \succeq r_3$.
3. $\forall r.r \succeq r$.

Exercice 5 : (Bonus) Ouverture (/+2)

A propos de la série Mr. Robot, on lit dans un blog intitulé "*4 Security Lessons We Can Learn From 'Mr. Robot*' " l'extrait suivant.

> **Become a Human Scam Detector**
>
> *Hackers like Elliot (ndlr : le héros de la série) often use Social Engineering attacks to compromise the human element. Human exploits can circumvent a lot of the technical security measures put in place to protect data. Most people's instinct is to help others and this is what Social Engineers like to capitalize on.*

1. Commentez en 50 mots maximum.