Université de Tlemcen	L3 Informatique
Faculté des Sciences	Module Sécurité Informatique
Département Informatique	Examen Final S2 Durée 1h30
Corrigé type	

Exercice 1 (6 pts, 1 pt par question)

Entourez la ou les bonnes réponses (une fausse réponse entourée entrainera une diminution de la note)

Q1) Un algorithme de chiffrement peut être utilisé dans la mise en œuvre des services de sécurité suivants

1) Confidentialité
2) Authentification
3) Intégrité

Q2) Une fonction de hachage H possède les propriétés suivantes

3) Génère une empreinte de taille fixe

Q3) L'authentification est basée sur le fait de pouvoir Prouver

1) Connaitre une information secrète
2) Détenir une information secrète
3) Etre en mesure de reproduire une information secrète

Q4) Le Chiffrement en mode bloc ECB est caractérisé par

4) Aucune des propriétés précédentes

Q5) le chiffrement en mode bloc CBC est caractérisé par :

4) Le déchiffrement des blocs ne peut pas se faire en parallèle

Q6) une vulnérabilité/faille dans une application ou un système peut se situer au niveau :

1) Conception
2) Configuration
3) Implémentation
4) Utilisation

Exercice 2 (5 pts)

Pour chacune des questions suivantes, choisissez ou donnez (case autre :) la bonne réponse, en Justifiant brièvement votre réponse

Q1) On désire chiffrer un message **M** d'une taille de 1906310 octets, en utilisant un chiffrement par bloc en mode CBC où la taille d'un bloc est de 96 bits. Le message chiffré **C** aura une taille de

4) Autre :1906320 (0. 5) . Le nombre de blocs en clair à chiffrer est 1906310/12=158859.166 donc 158860 blocs. La taille du message chiffré est 158860*12= 1906320 bytes (0.5)

Q2) En l'absence d'un attaquant/attaque, pour garantir l'intégrité d'un message **M** envoyé entre deux extrémités

1) Il suffit de générer un haché sur M en utilisant une fonction de hachage (0.75)

En l'absence d'un attaquant, seules les erreurs de transmission (involontaires) peuvent affecter M et dans ce cas le haché sur M (h =H(M)) suffit pour garantir son intégrité. Il n'y a donc pas d'attaquant changeant volontairement M et calculant un nouveau haché sur M. la probabilité qu'une erreur aléatoire affecte M le modifiant en M' et en parallèle affecte h le modifiant en h' où on aura h'=H(M') est pratiquement nulle (1.25)

Q3) l'authentification Linux et/ou Windows utilisée pour authentifier des utilisateurs possédant des comptes locaux, peut aussi être utilisé sans risque pour une authentification en réseau

2) Faux (0.5)
   L'authentification en local Windows/Linux étant sujet à une attaque de type rejeu (replay), on ne peut pas l'utiliser sans risque pour une authentification en réseau (0.5)

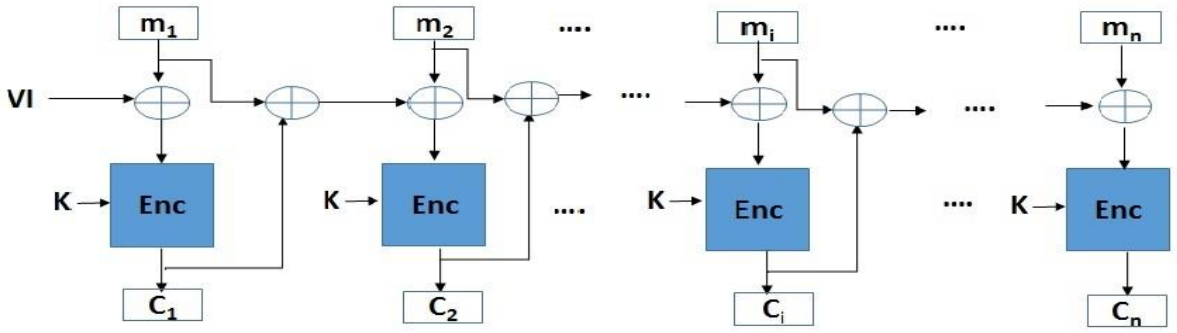
Q4) La cryptographie est fondée sur le fait que les algorithmes utilisés (chiffrement, hachage) soient secrets

2) Faux (0.5)

Les algorithmes (chiffrement, hachage, etc.) sont publique, leur spécification / implémentation connue et documentée. C'est plutôt les clés utilisées qui doivent être gardés secrets (0.5)

Exercice 3 (5 pts)

On désire étudier le mode de chiffrement en bloc PCBC (Propagating CBC) identifié dans le schéma suivant



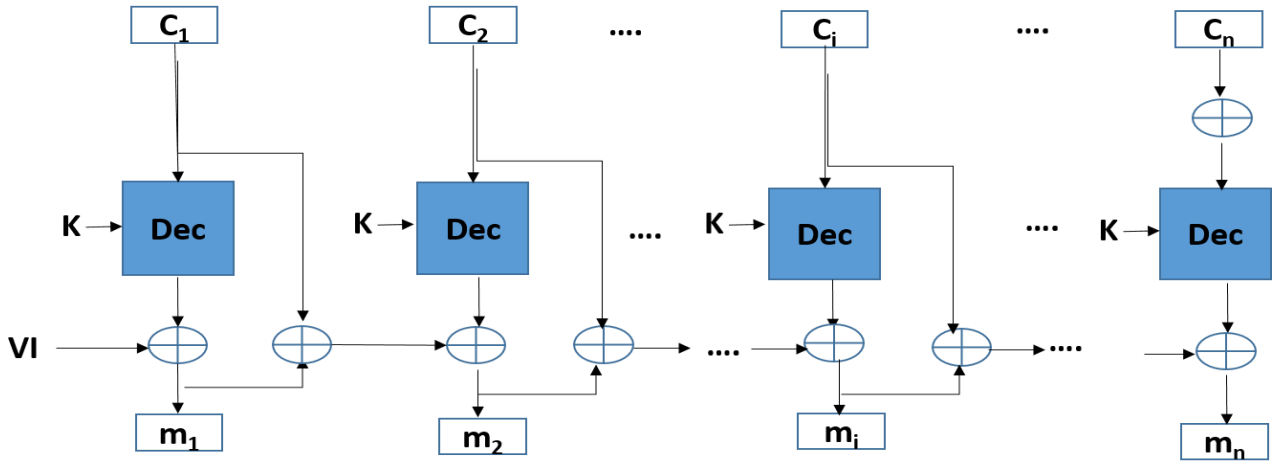
$\mathbf{m_i}$ représente un bloc en clair, $\mathbf{C_i}$ le bloc chiffré correspondant, **Enc** une fonction de chiffrement, et **K** la clé de chiffrement, ⊕ : XOR

Q1) Donnez l'expression pour chaque bloc chiffré $\mathbf{C_i}$ **(1 pt)**

C_i= Enc(K, m_i ⊕ m_{i-1}⊕ C_{i-1})

Avec C_0= VI et m_0=0

Q2) Déduire le schéma de déchiffrement du mode PCBC en donnant l'expression pour chaque bloc clair déchiffré $\mathbf{m_i}$ **(0.75 + 0.75)**



m_i= Dec(K, C_i) ⊕ m_{i-1} ⊕ C_{i-1}

avec **m_0 =0 et C_0=VI**

Q3) Est-ce que le mode PCBC peut être parallélisable en chiffrement et/ou déchiffrement ? Justifiez
Non il n'est pas parallélisable car le chiffrement d'un bloc clair $\mathbf{m_i}$ dépend du bloc chiffré précédent $\mathbf{C_{i-1}}$ et le déchiffrement du bloc $\mathbf{C_i}$ dépend du bloc en clair précédent $\mathbf{m_{i-1}}$ **(0.5** +0.75)

Q4) Quel est l'impact de la perte et/ou modification d'un bloc chiffré $\mathbf{C_i}$?
En cas de perte d'un bloc chiffré $\mathbf{C_i}$ on ne pourra pas déchiffrer les blocs $\mathbf{C_j}$ avec j>i
Si une modification se produit au niveau d'un bloc $\mathbf{C_i}$ on ne pourra pas déchiffrer correctement $\mathbf{C_i}$ en $\mathbf{m_i}$ et par conséquent, on sera incapable de déchiffrer correctement les blocs suivants $\mathbf{C_j}$ en $\mathbf{m_j}$ (j>i) (1.25)

Exercice 4 (4 pts)

Soit le protocole d'authentification suivant se déroulant entre deux extrémités **A**, **B** où $\mathbf{K_{AB}}$ est un secret pré-partagé entre A et B :

1) A → B: ID_A, N_A
2) B → A: ID_B, $\{ID_B||ID_A||N_B||H(N_A||N_B)\}\_K_{AB}$
3) A → B: ID_A, $\{ID_A||ID_B||H(N_B||N_A)\}\_K_{AB}$

Note: ID_X : identité de l'extrémité X X →Y message envoyé de X à Y u||v : concaténation de u et v

{M}_K : le message M chiffré en utilisant la clé K $\mathbf{N_X}$: une valeur aléatoire unique générée par X

H : fonction de hachage

Q1) Qu'elle est l'utilité des valeurs N_A et N_B ? (1 pt)

Elles protègent contre les attaques par rejeu de paquet : N_A protège A contre le rejeu de 2) et N_B protège B contre le rejeu de 3). Aussi ils représentent la question (Challenge) que va envoyer chaque extrémité à l'autre.

Q2) Est-ce que ce protocole permet une authentification à un seul sens ou mutuelle ? Justifiez (0.25 + 0.75)

Une authentification mutuelle, car en vérifiant le message 2) A peut s'assurer de l'identité de B et en vérifiant le message 3) B peut s'assurer de l'identité de A où chaque extrémité prouve connaitre K_{AB}

Q3) Est ce que ce protocole est cible à une attaque de type rejeu de paquet (replay attack) ? Justifiez (0.5+0.5)

Non il n'est pas cible, et ceci grâce à l'utilisation des valeurs aléatoires uniques N_A et N_B

Q4) Supposant que A et B n'ont aucun moyen pour pré-partager le secret K_{AB}, comment pourront-il dans ce cas s'authentifier ?

Il leur suffit d'utiliser un protocole d'authentification basé sur un serveur d'authentification de confiance comme Needham-Schroedr ou Kerberos.