M1 informatique – UE Réseau et sécurité
Année universitaire : 2020-2021
Durée : 2h
Consignes : Documents interdits (sauf dictionnaires papiers). Calculatrices et portables interdits.

Examen de réseau et sécurité – première session

Exercice 1 – 3-DES (4 points)

L'algorithme 3-DES est un algorithme se basant sur DES qui est plus robuste que DES aux attaques. L'algorithme 3-DES consiste à chiffrer un message en appliquant trois fois successivement l'algorithme DES. Si $f(M,k)$ désigne le chiffrement de M par DES avec la clé k, alors l'algorithme 3-DES consiste à calculer $f(f(f(M,k_1),k_2),k_3)$.
Question 1.1 (1 point) : Quelle est la taille d'une clé 3-DES (version non optimisée) ? Quelle est donc la complexité pour casser 3-DES en force brute ?
Question 1.2 (1 point) : Quelle est la complexité pour casser 3-DES avec la technique *meet-in-the-middle* ? Vous utiliserez un dessin pour justifier votre réponse.

Comme la complexité pour casser 3-DES (cf question 1.2) est inférieure à la taille de clé de 3-DES (cf question 1.1), des chercheurs ont proposé de réduire la taille de la clé de 3-DES, ce qui évite un faux sentiment de protection. Ils proposent que parmi les trois clés k_1, k_2 et k_3, deux clés soient identiques.
Question 1.3 (1 point) : Montrez que si $k_1=k_2$ (ou $k_2=k_3$), alors l'algorithme 3-DES se casse facilement.
Question 1.4 (1 point) : Montrez que si $k_1=k_3$ (k_2 étant différente), alors la complexité pour casser l'algorithme 3-DES est égale à la taille (optimisée) de la clé de 3-DES.

Exercice 2 – Cryptanalyse différentielle (5 points)

La figure 1 représente la S-box S5 de DES. Pour rappel, $S_5(110110)=5_{10}$ car l'entrée *110110* correspond à la ligne *1----0* et à la colonne *-1011-*.

S_5	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
00	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
01	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
10	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
11	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3

Figure 1 : La S-box S5 de DES.

Question 2.1 (1 point) : Quelle est la valeur de $S_5(100101)$, en décimal ?
Question 2.2 (3 points) : Nous allons maintenant essayer de cryptanalyser S5. Pour cela, nous générons deux messages M_1 et M_2, tels que $I_1 \oplus I_2 = 100000$. Nous observons que $O_1 \oplus O_2 = 1111$. Quels sont les couples (I_1,I_2) valides ? Pour répondre à la question, utilisez la table de la figure 2 (il n'est pas nécessaire de remplir toutes les cases, typiquement pour I_2 ou parfois pour $O_1 \oplus O_2$).
Question 2.3 (1 point) : Déduisez-en les clés possibles, sachant que $E_1=110100$? Rappelons que $I_1=E_1 \oplus K$.

Exercice 3 – le réseau TOR (11 points)

TOR (en anglais, le routeur oignon) est un protocole réseau permettant d'anonymiser des connexions, en rendant difficile la surveillance du trafic.
Question 3.1 (1 point) : Sur Internet, qui peut techniquement surveiller le trafic d'un utilisateur donné ?
Question 3.2 (1 point) : Pourquoi est-il important pour certains utilisateurs (des journalistes, des défenseurs des droits de l'Homme, des personnes opprimées) d'être anonymes sur Internet ?

TOR est un réseau de superposition, ce qui signifie que TOR s'exécute au-dessus d'Internet. Pour établir une route entre une source s et une destination d, s construit un chemin aléatoire de routeurs TOR (r_1, r_2, ..., $r_n=d$). Le message envoyé de s à d est chiffré de manière que chaque routeur intermédiaire r_i ne connaisse que r_{i-1} et r_{i+1}, mais ni s (sauf pour r_1) ni d (sauf pour r_{n-1}). De plus, seul $r_n=d$ peut déchiffrer le message.
Question 3.3 (3 points) : Décrivez la procédure de chiffrement/déchiffrement appliquée par s, en incluant le type de clés que vous proposez (symétriques ou asymétriques).
Question 3.4 (1 point) : Dans la spécification de TOR, il est dit que durant la construction du chemin aléatoire, le premier routeur r_1 doit être un nœud gardien. Qu'est-ce que pourrait être un nœud gardien ? Pourquoi TOR spécifie cette contrainte ?

Pour permettre aux clients de se connecter à lui, l'adresse IP d'un serveur doit être connue des clients. Cependant, dévoiler l'adresse IP d'un serveur lève son anonymat. TOR propose donc le protocole suivant. (1) Le serveur choisit un routeur aléatoire et lui demande de devenir son serveur relais, dont le rôle est d'accepter les connexions entrantes. (2) Le serveur publie l'adresse du serveur relais (via un forum par exemple). (3) Le client choisit un routeur aléatoire et lui demande de devenir un point de rendez-vous. (4) Le client informe le serveur relais du point de rendez-vous. (5) Le serveur relais transmet la requête au serveur. (6) Le serveur se connecte au point de rendez-vous, et les communications ultérieures entre le client et le serveur passent par ce point de rendez-vous. Toutes les connexions de cette description sont faites en utilisant des chemins TOR, pour maintenir l'anonymat des deux extrémités.
Question 3.5 (1 point) : Justifiez les propriétés suivantes :

- Le client ne connaît pas l'adresse du serveur, mais connaît l'adresse du point de rendez-vous.
- Le serveur ne connaît pas l'adresse du client, mais connaît l'adresse du point de rendez-vous.

Question 3.6 (1 point) : Justifiez pourquoi le point de rendez-vous ne connaît ni l'adresse du client ni celle du serveur.

L. Øverlier et P. Syverson ont proposé une attaque dans [1] pour déterminer la localisation de serveurs sur TOR. L'attaquant doit contrôler un routeur TOR (ce qui est simple, étant donné que TOR se base sur le volontariat) et un client. Le but de l'attaquant est d'être élu comme le premier routeur TOR du chemin entre le serveur et le point de rendez-vous. L'attaquant essaye de faire correspondre le trafic envoyé par le client avec le trafic du routeur contrôlé. Ensuite, il utilise des informations de délai pour déterminer à quel endroit du chemin il se trouve.
Question 3.7 (1 point) : Pourquoi est-il important d'être élu comme le premier routeur du chemin entre le serveur et le point de rendez-vous ?
Question 3.8 (1 point) : Quel est l'intérêt de l'analyse de trafic ? Comment marche-t'elle ?
Question 3.9 (1 point) : Expliquez la partie de l'attaque utilisant les informations de délai.

Référence :
[1] L. Øverlier, P. Syverson. "Locating Hidden Servers", IEEE Symposium on Security and Privacy, 2006.

Nom : **Prénom :**

I_1	I_2	O_1	O_2	$O_1 \oplus O_2$	Valide ?	I_1	I_2	O_1	O_2	$O_1 \oplus O_2$	Valide ?
000000						100000					
000001						100001					
000010						100010					
000011						100011					
000100						100100					
000101						100101					
000110						100110					
000111						100111					
001000						101000					
001001						101001					
001010						101010					
001011						101011					
001100						101100					
001101						101101					
001110						101110					
001111						101111					
010000						110000					
010001						110001					
010010						110010					
010011						110011					
010100						110100					
010101						110101					
010110						110110					
010111						110111					
011000						111000					
011001						111001					
011010						111010					
011011						111011					
011100						111100					
011101						111101					
011110						111110					
011111						111111					

Figure 2 : Tableau de cryptanalyse différentielle de S5.