Réseaux 6e édition

Andrew Tanenbaum, Nick Feamster, David Wetherall

Corrigés des exercices

Chapitre 8 : Sécurité des réseaux

ISBN : 978-2-3260-0239-5

1. En vérifiant avant chaque opération si elle est autorisée, on réduit les performances du système.
2. Les sommes CRC servent à détecter les erreurs isolées et celles en rafale. Elles ne sont pas conçues et ne sauront pas détecter des altérations frauduleuses du message. L'attaquant peut injecter son contenu puis remplacer le CRC par celui qu'il calcule.
3. La lettre **F** désigne un paquet FIN, donc c'est un scan FIN sur le port 21 de plusieurs hôtes. La réponse R ACK provenant de *host102.caesar.org* indique que son port 21 est fermé. Il est crucial de savoir que cet hôte existe et qu'il est en ligne. La fonction scanner ne peut pas savoir si les autres hôtes exécutent le service FTP si elle ne sait pas si ces machines existent et si la fonction n'a pas vu de message d'erreur ICMP. Il est conseillé de tenir compte de ces points avant tout jugement à propos d'un événement réseau que vous étudiez. Vous remarquez que *brutus* envoie ses scans depuis son port 53 (celui du DNS !), sans doute pour tromper les pare-feu. Notez les longs silences entre les scans (mode furtif) !
4. Un sondage de Noël (Xmas) des ports 22 (ouvert), 25 (fermé), 80 (ouvert) et 10 000 (fermé).
5. i) tout est compromis ; ii) tout est compromis ; iii) rien n'est compromis.
6. Rappel = $R = TP/(TP + FN) = 40 / 70 = 4/7 = 0{,}57$

   Précision = $P = TP/(TP + FP) = 40 / 50 = 0{,}8$

   Mesure-F = $2PR/(P + R) = 0{,}67$

   Exactitude (Accuracy) = $A = (TP + TN)/(TP + TN + FP + FN) = (999920 + 40)/10\hat{}6 = 0{,}99996$.
7. L'oubli de fréquence de base est un piège et vous pouvez vous attendre à ce que les personnes pensent souvent à tort qu'une alerte est très probable. On peut le prouver avec un peu de calcul autour du théorème de Bayes. Dans l'exemple de test avec des malades et des personnes en bonne santé, nous avions posé que :

   $$P(S|Pos) = \frac{P(S)P(Pos|S)}{P(Pos)}$$

   Au lieu d'un M pour les Malades, nous utilisons ici un I (pour Incident) et au lieu d'un B pour les gens en Bonne santé, nous optons pour le non-incident NOT(I). Nous remplaçons POS par A (pour Alarme) et NEG par NOT(A). Nous pouvons ensuite peupler la formule :

   $$P(I|A) = \frac{P(I)P(A|I)}{P(A)} = \frac{P(I)P(A|I)}{P(I)P(A|I) + \big(1 - P(I)\big)P(A|NOT(I))}$$

   La formule est dominée par le taux de faux positifs $\boldsymbol{P(A|NOT(I))}$. Il semble donc que la détection fonctionne bien, bien que la plupart des alertes soient fausses.
8. Connectez-vous au serveur et envoyez un paquet falsifié, provenant prétendument de la machine de Hubert puis envoyez 100 faux paquets RST de votre machine vers le serveur. Comptez le nombre de ACK de défi challenge que vous recevez.

9. C'était le début de la fable « Le corbeau et le renard » :

   ```
   ockvtg eqtdgcw  uwt wp ctdtg rgtejg
   maître corbeau sur un arbre perché
   vgpckv hp uqp dge wp htqocig
   tenait en son bec un fromage
   ockvtg tgpctf rct nqfgwt  cnngejg
   maître renard par l'odeur alléché
   nwk vkpv c rgw rtgu eg ncpicig
   lui tint à peu près ce langage
   ```

10. La source était un extrait de la première phrase de la section 1.1 de la page 1 de ce livre.Voici le texte déchiffré :

    ```
    l'industrie informatique a accompli des progrès fantastiques
    ```

    Le mot-clé est ESCARGOT, donc de longueur 8.

11. Puisqu'elle possède la clé du chiffre de transposition chiffrée, Ève connaît sa longueur. Elle sait donc combien de colonnes il y avait dans la matrice de transposition et peut ainsi répartir le texte chiffré sur ces colonnes. Tout ce qu'il lui reste alors à faire pour déchiffrer le message est d'essayer tous les arrangements de colonnes jusqu'à ce qu'elle en trouve un avec lequel le message a un sens. Si la longueur de la clé chiffrée est de k caractères, il lui faudra au plus 2^k essais.

12. Le texte chiffré serait le même quel que soit l'ordre de chiffrement parce que le chifre par substitution et celui par transposition modifient différents attributs du texte source. La substitution conserve l'ordre des lettres en les remplaçant par d'autres. La transposition change l'ordre des lettres sans modifier leur représentation.

13. Voici ce masque jetable :

    1010011 0001110 1100010 1010110 1001011 0100110 1111100 0111100 1001010 1111111

14. Pour chiffrer vos messages, vous pouvez utiliser la représentation en ASCII des caractères composant *Le Seigneur des Anneaux*. Cela vous donne un masque jetable du nombre de bits nécessaires à la représentation de tous les caractères composant cet ouvrage. Lorsque vous arrivez vers la fin du livre, vous utilisez la dernière partie pour envoyer un message donnant le titre de l'ouvrage qui va vous servir pour les messages suivants. Et ainsi de suite. Comme vous avez un nombre infini de livres, votre masque jetable est infiniment long.

15. À 250 Gbit/s, il faut 4×10^{-12} seconde pour transmettre un bit. La vitesse de la lumière dans la fibre étant de 2×10^8 m/s, l'impulsion lumineuse parcourt 0,8 mm ou 800 microns dans un temps bit. Un photon prenant une longueur de 1 micron, l'impulsion lumineuse est longue de 800 photons. Nous sommes loin d'avoir un photon par bit. Ce n'est qu'à 200 Tbit/s que nous atteindrons 1 bit par photon.

16. Dans la moitié des cas, Ève fait la bonne hypothèse. Tous ces bits sont alors regénérés correctement. Dans l'autre moitié des cas, elle se trompe et envoie à Bob des bits aléatoires, dont la moitié sont faux. Donc, au total, 25 % de bits qu'elle met sur la fibre sont faux. Le masque jetable de Bob contiendra 25 % de bits en erreur.

17. Si l'intrus a une puissance de calcul infinie, les deux formes sont équivalentes. Mais, en pratique, ce n'est pas le cas et la seconde forme est alors meilleure. Elle oblige, en effet, l'intrus à faire un calcul pour voir si chaque clé essayée est correcte. Si ce calcul est complexe, il ralentit considérablement l'intrus.

18. Une somme de contrôle de type CRC permettrait d'ajouter de la redondance au message et une horodate de pouvoir s'assurer de son actualité.

19. Plusieurs approches sont possibles. On peut chercher à exploiter les points faibles d'un autre système servant à la fonction de messagerie instantanée. Par exemple, son système d'exploitation peut souffrir d'une faiblesse que la police peut exploiter pour accéder aux messages déchiffrés. Une autre approche vise la sécurité physique par exemple avec un agent double qui soutire à quelqu'un une clé pour déchiffrer le texte sans avoir à chercher) le décrypter.

20. L'équation $2^n = 10^{16}$ nous donne n, le nombre de périodes de doublement nécessaires. On en tire $n = 16 \log_2 10$ ou encore $n = 53{,}15$ périodes de doublement, soit 79,72 ans. Mais il n'est pas certain que la loi de Moore s'applique encore pendant près de 80 ans !

21. Il nous faut résoudre l'équation $2^{256} = 10n$. En passant aux logarithmes, on en déduit $n = 256\log 2$, soit $n = 77$. Le nombre de clés est ainsi de 10^{77}. Le nombre d'étoiles de notre galaxie est de 10^{12}, et il y a probablement 10^8 galaxies dans l'univers, ce qui conduit à un nombre total d'étoiles d'environ 10^{20}. La masse du soleil, étoile bien caractéristique, est de 2×10^{33} g. Le soleil est composé essentiellement d'hydrogène et le nombre d'atomes dans 1 g d'hydrogène est environ de 6×10^{23} (nombre d'Avogadro). Le nombre d'atomes du soleil est donc de $1{,}2 \times 10^{57}$. Si l'on a 10^{20} étoiles, le nombre d'atomes de toutes les étoiles de l'Univers est à peu près égal à 10^{77}. On le voit : le nombre de clés AES 256 bits est égal au nombre d'atomes de l'Univers tout entier (sans tenir compte de la matière noire). Ce n'est donc pas demain qu'on cassera AES-256 par une attaque directe.

22. Malheureusement, tous les blocs de texte en clair à partir de P_{i+1} seront erronés puisque toutes les entrées des boîtes XOR seront fausses. Une telle erreur est donc beaucoup plus grave qu'une inversion de bit.

23. Le chiffrement par chaînage de blocs produit 8 octets en sortie par action de chiffrement alors que la rétroaction n'en produit qu'un. Il est donc huit fois plus efficace (avec le même nombre de cycles, on peut chiffrer huit fois plus de texte en clair).

24. Bob peut lire le message en le déchiffrant d'abord avec sa clé privée (ce qui produit $D_A(P)$) puis en le chiffrant avec la clé publique de Alice (ce qui produit P).

25. L'effort restant est presque égal à zéro ! Aussi bien e que n sont publics et premiers entre eux. Au moyen de la version étendue de l'algorithme d'Euclide, on trouve facilement d, et on peut reconstruire la clé privée.

26. Maria doit songer à changer ses clés parce que Francis pourra assez facilement trouver sa clé privée. Francis possède déjà la clé publique de Maria ($e1$, $n1$) et remarque que $n2 = n1$. Francis peut alors deviner la clé privée de Maria (*d1, n1*) en essayant les différentes solutions de l'équation $d1 \times e1 = 1 \; mod n1$.

27. Le R_As du dernier message est peut-être encore en mémoire RAM. S'il a disparu, Ève peut essayer de renvoyer le plus récent message à Bob, en espérant qu'il ne détecte pas que c'est un doublon. Une solution pour Bob consiste à écrire le R_a de chaque message entrant sur disque *avant* autre chose. L'attaque par rejeu ne pourra alors plus réussir. Mais un autre danger apparaît : si un plantage se produit pendant qu'une requête est écrite sur disque, elle ne sera jamais exécutée.

28. Si Ève remplace à la fois P et la signature, lorsque Bob appliquera la clé publique d'Alice à la signature, il obtiendra un élément qui ne sera pas le condensat du texte en clair. Ève peut remplacer un message et en faire le hachage mais elle ne peut le signer avec la clé privée d'Alice.

29. Lorsqu'un client, appelons-le Marcel, fait savoir qu'il veut dialoguer sur une messagerie rose ou jouer dans un casino virtuel, la mafia commande au joaillier un diamant en donnant le numéro de la carte de crédit de Marcel. Dès que le joaillier envoie son contrat à signer (contrat donnant le numéro de carte de crédit et un numéro de boîte postale en guise d'adresse de la mafia), la mafia retransmet le hachage du message du joaillier à Marcel, en même temps que le contrat faisant de Marcel un client d'un site rose ou d'un casino virtuel. Si Marcel se contente de signer sans vérifier que le contrat et la signature ne concordent pas, la mafia fait suivre la signature au joaillier qui expédie alors le diamant. Marcel pourra toujours dire qu'il n'a jamais commandé de diamant, il sera facile au joaillier de produire un contrat signé montrant qu'il l'a bien fait.

30. Vingt-cinq étudiants permettent de faire $(25 \times 24)/2 = 300$ paires d'étudiants. La probabilité que dans une paire quelconque, les deux étudiants aient la même date d'anniversaire est de 1/181 et la probabilité qu'ils aient une date différente est de 180/181. La probabilité que les 300 paires aient une date différente est donc de $(180/181)^{300}$, soit environ 0,190. Si la probabilité que toutes les paires aient une date différente est de 0,190, la probabilité qu'il y en ait au moins une de même date est de 0,810.

31. La secrétaire peut saisir un certain nombre d'espaces (par exemple 32) puis remplacer chacun d'eux par espace, retour arrière, espace. Sur le terminal, toutes les variantes paraîtront identiques mais correspondront à des condensats différents. Ainsi, l'attaque de type anniversaire fonctionnera. Autres méthodes : ajouter des espaces à la fin du message ou placer des tabulations à la place des espaces.

32. Oui, c'est possible. Alice chiffre un nonce avec la clé partagée et l'envoie à Bob. Bob renvoie alors un message chiffré avec la clé partagée, message contenant le nonce, son propre nonce et la clé publique. Ève ne peut pas fabriquer un tel message et si elle envoie un message généré au hasard, ce dernier ne contiendra pas le nonce d'Alice. Pour compléter le protocole, Alice renvoie le nonce de Bob chiffré avec la clé publique de Bob.

33. La première étape consiste à vérifier le certificat X.509 en utilisant la clé publique du *root* de CA. S'il est authentique, Alice possède maintenant la clé publique de Bob, à condition d'avoir consulté la liste des certificats révoqués, s'il en existe une. Mais pour savoir si Bob est bien à l'autre bout, elle doit savoir si ce dernier possède la clé privée correspondante. Elle sélectionne donc un nonce et le lui envoie chiffré avec sa clé publique. Si Bob peut le renvoyer en clair, elle est sûre que c'est bien lui.

34. Alice doit tout d'abord établir un canal de communication avec *X* pour lui demander un certificat afin de vérifier sa clé publique. Supposons que *X* fournisse un tel certificat signé par un autre CA : *Y*. Si Alice ne connaît pas *Y*, elle répète le processus précédent, jusqu'à ce qu'elle reçoive un certificat vérifiant la clé publique du CA *Z* signé par *A* dont Alice connaît la clé publique. Notez que cela peut continuer jusqu'à la racine (*A*, dans ce cas, est la racine). Puis Alice vérifie les clés publiques en ordre inverse, en partant du certificat fourni par *Z*. À chaque étape de la vérification, elle accède également à la liste des certificats révoqués pour être certaine que les certificats fournis sont bien valides. Enfin, après vérification de la clé publique de Bob, Alice s'assure que c'est bien lui qui est de l'autre côté en utilisant la même méthode qu'à l'exercice précédent.

35. Non. AH en mode transport inclut l'en-tête IP dans la somme de contrôle. Le boîtier NAT change l'adresse source, ce qui modifie la somme de contrôle. Tous les paquets vont donc être considérés comme erronés.

36. Les HMAC se calculent beaucoup plus rapidement que RSA et sont donc à recommander. Cependant, cela implique d'établir une clé partagée avec Bob avant la transmission du message.

37. Les HMAC se calculent beaucoup plus rapidement.

38. Il faut traiter le trafic entrant, ne serait-ce qu'en raison de la présence éventuelle de virus. Pour ce qui est du trafic sortant, on peut voir s'il n'y a pas de fuites d'informations confidentielles. Le test des virus est réalisé assez efficacement avec un bon antivirus. Mais le test du trafic sortant, surtout s'il est chiffré, est très difficile à réaliser et risque d'être peu efficace.

39. Si Jean veut que personne (y compris son administrateur système) ne connaisse la nature de ses communications, il doit utiliser des mécanismes de sécurité complémentaires. Un VPN, en effet, n'assure la sécurité des communications que vis-à-vis d'Internet (pour discuter avec Marie de projets confidentiels de R&D, par exemple) mais pas vis-à-vis de l'intérieur de l'organisation (pour discuter de l'augmentation de salaire qu'elle lui a promis). Si Jean ne se sent menacé que par des gens extérieurs à l'organisation, le VPN est suffisant.

40. Dans le message 2, placez R_B dans le message chiffré et non à l'extérieur. De cette façon, Ève ne peut découvrir R_B et l'attaque par réflexion ne fonctionne pas.

41. Bob sait que $g^x \bmod n = 82$. Il calcule $82^3 \bmod 227 = 155$. Alice sait que $g^y \bmod n = 125$. Elle calcule $125^{12} \bmod 227 = 155$. La clé est 155. La façon la plus simple de réaliser ces calculs est d'utiliser le programme Unix *bc*.

42. a) L'information passée par Alice à Bob n'est pas chiffrée, par conséquent il n'y a rien que Bob ne connaisse et qu'Ève ne connaisse pas. Toute réponse que Bob peut donner peut également être donnée par Ève. Ainsi, il est impossible pour Alice de savoir si elle dialogue avec Bob ou avec Ève.

    b) Si *n* ou *g* sont secrets et inconnus d'Ève, celle-ci ne peut prétendre être Bob puisqu'elle est incapable d'exécuter les calculs corrects pour envoyer un message de retour à Alice et obtenir ainsi la clé correcte.

43. Le KDC doit savoir qui lui envoie un message pour appliquer la bonne clé de déchiffrement.

44. Ils servent à l'authentification en empêchant les attaques par rejeu.

45. Non. Il suffit que Ève capture deux messages de ou pour le même utilisateur puis tente de les décrypter avec la même clé. Si le champ du nombre aléatoire est le même dans les deux, elle a trouvé la bonne clé ! Cette précaution ne fait que doubler sa charge de travail.

46. La cryptographie à clé publique assure la confidentialité en chiffrant les messages avec la clé publique du destinataire. Elle assure intégrité, authentification et non-répudiation par signature du message.

47. Les condensats assurent intégrité, authentification et non-répudiation s'ils sont utilisés dans la signature des messages. Les attaquants ne doivent pas pouvoir remplacer le condensat par le leur. Il suffit par exemple de faire chiffrer le condensat par l'expéditeur.

48. Les deux nombres aléatoires sont utilisés dans des buts différents. R_A sert à convaincre Alice qu'elle parle bien au KDC. R_{A2} sert à convaincre Alice qu'elle parle bien à Bob. Les deux sont donc nécessaires.

49. Si AS est défaillant, de nouveaux utilisateurs pourtant légitimes ne seront plus capables de s'authentifier eux-mêmes, c'est-à-dire d'obtenir un ticket TGS. Les utilisateurs qui ont déjà un ticket TGS (obtenu avant la panne de AS) peuvent continuer à accéder aux serveurs jusqu'à la fin de validité de leur ticket. Si le serveur de vérification de ticket TGS est défaillant, seuls les utilisateurs ayant déjà obtenu un ticket de serveur (donné par TGS avant sa panne) pour le serveur S pourront accéder à S jusqu'à la fin de validité de leur ticket. Dans les deux cas, il n'y a pas de violation des règles de sécurité.

50. Même si Ève intercepte le message contenant R_B, elle n'a aucun moyen de l'utiliser puisque sa valeur ne sera plus reprise dans la communication entre Alice et Bob. Il n'y a donc pas besoin qu'Alice et Bob répètent le protocole avec des valeurs différentes pour garantir la sécurité de leurs communications. Mais Ève pourrait utiliser l'information glanée avec ce message intercepté (et avec de nombreux autres messages interceptés) pour essayer de comprendre comment Bob engendre ses nombres aléatoires. C'est pourquoi, la prochaine fois, Alice ne devrait quand même pas oublier de chiffrer le dernier message du protocole.

51. Il n'est pas essentiel de chiffrer R_B avant de l'envoyer. Ève n'a aucun moyen de le connaître et il ne sera plus réutilisé, donc il n'est pas réellement secret. D'un autre côté, chiffrer permet de faire un essai avec K_S pour être sûr que tout se passe bien avant d'envoyer les données. Et puis, pourquoi donner à Ève des informations sur le générateur de nombres aléatoires de Bob ? En général, il est préférable d'envoyer en clair le moins d'éléments possibles, surtout si (comme ici) le coût du chiffrement est faible.

52. La banque envoie un défi (un nombre aléatoire très long) à l'ordinateur du commerçant qui le transmet à la carte. L'unité centrale (UC) de la carte le transforme alors en fonction du code individuel (code PIN) entré directement dans la carte. Le résultat de cette transformation est fourni à l'ordinateur du commerçant pour transmission à la banque. Si le commerçant rappelle la banque pour une nouvelle transaction, celle-ci enverra un nouveau défi et tout ce que le commerçant aura pu apprendre avec l'ancien lui sera inutile. Même si le commerçant parvient à connaître l'algorithme utilisé par la carte, il ne connaît pas le code PIN de l'utilisateur puisque celui-ci est entré directement dans la carte. L'affichage sur la carte est nécessaire pour éviter que le commerçant dise que le prix est de 49,95 mais qu'il envoie 499,95 à la banque.

53. Le premier email était un hameçonnage dont l'auteur avait changé le champ FROM: pour faire croire qu'il provenait de la banque. L'adresse URL insérée menait vers un site contrôlé par l'attaquant, simulant l'aspect du site de la banque. En vous connectant, vous lui fournissiez vos crédentiels/ L'attaquant n'avait plus qu'à s'en servir sur le vrai site de la banque pour virer votre solde positif.

54 La compression réduit le volume à transférer, mais surtout, elle rend le contenu beaucoup plus difficile à déchiffrer car les indices de fréquence des lettres ont disparu (le « e » est la lettre la plus fréquente en français comme en anglais). Le texte devient incompréhensible, ce qui décuple les efforts que doivent produire les cryptanalystes pour le lire.

55. Pour envoyer un message PGP en multicast, il faudrait chiffrer la clé IDEA avec la clé publique de chacun des utilisateurs de l'adresse internet. Si tous les utilisateurs auxquels on veut envoyer un message en multicast ont la même clé publique, alors on peut effectivement le faire.

56. Non. Supposons que l'adresse soit celle d'une liste de diffusion. Chaque personne possède sa propre clé publique. Chiffrer la clé IDEA avec une seule clé publique ne fonctionnera pas. Il faudrait la chiffrer avec des clés publiques multiples.

57. Le nonce protège des attaques par rejeu. Puisque chaque partie contribue à la clé, si un intrus essaie de rejouer d'anciens messages, la nouvelle clé générée ne correspondra pas à l'ancienne.

58. L'image compte 2 048 × 512 pixels. Puisque chaque pixel contient 3 bits de poids faible, on peut utiliser pour la stéganographie 2 048 × 512 × 3 soit 3 145 728 bits ou encore 393 216 octets. On chiffre ainsi dans l'image environ 16 % du fichier. Si le fichier avait été comprimé au quart de sa taille d'origine, il ne ferait plus que 0,625 Mo. On pourrait alors en cacher environ 63 % dans l'image.

59. C'est simple. La musique n'est qu'un fichier comme un autre. On peut placer les 294 912 octets dans les bits de poids faible. MP3 nécessitant environ 1 Mo par minute, on peut placer dix-huit secondes de musique.

60. Le nombre de bits à chiffrer est de $60 \times 10^6 \times 8 = 480 \times 10^6$ bits. Chaque pixel de l'image peut cacher 3 bits. Pour qu'une image contienne tout le fichier, il faut qu'elle fasse $480 \times 10^6/3$ pixels, soit $160 \times 10^6 = 160\,000\,000$ pixels. Nous voulons une image de format 3:2, c'est-à-dire de largeur $3x$ et de hauteur $2x$. Le nombre de pixels est alors $6x^2$ ce qui nous donne pour x 5 164 et une image de 15 492 × 10 328 pixels. Si le fichier était comprimé au tiers de sa taille originelle, le nombre de bits à chiffrer serait de 160×10^6 et le nombre de pixels nécessaires de 53 333 333. L'image ferait alors 8 946 × 5 962.

61. Alice hache chaque message et le signe avec sa clé privée. Puis elle ajoute le hachage signé et sa clé publique au message. Les destinataires peuvent vérifier la signature et comparer la clé publique avec celle utilisée précédemment. Si Ève essaie de prendre la place d'Alice et ajoute la clé publique d'Alice, elle ne pourra pas mettre le bon hachage. Si elle ajoute sa propre clé publique, les gens s'apercevront que la clé publique a changé.

62 Les concepteurs n'ont pas assez obéi à plusieurs principes : le principe de simplicité, le principe de minimisation des mécanismes partagés et le principe de conception ouverte. Les micro-processeurs d'aujourd'hui sont des équipements très complexes, ce qui procure une grande surface d'attaque. Faire un seul espace mémoire à partir de celui du noyau et de celui des applications et partager les données des caches sont des mécanismes partagés à éviter. Enfin, la description détaillée des processeurs du commerce n'est pas disponible aisément, ce qui rend difficile les investigations des experts en sécurité qui cherchent à trouver les points faibles de ces processeurs. Les deux premiers principes sont bafoués pour augmenter les performances et le troisième est d'empêcher les concurrents de copier la conception des processeurs.

63 Le serveur n'est pas authentifié parce que vous ouvrez la session seulement avec un mot de passe, mais sans identifiant. Un attaquant peut activer un autre réseau Wifi portant le même nom que celui de l'hôtel ou bien un autre client de l'hôtel peut déduire votre clé de la sienne puis décrypter vos messages.

64. Les questions 64 à 68 sont des travaux pratiques à faire éventuellement évaluer par un enseignant ou un expert.