

[Tableau de bord](#) / [Mes cours](#) / [INF4420A - Sécurité informatique](#) / [Examen final](#) / [Examen Final A2020](#)

**Commencé le** samedi 19 décembre 2020, 13:40

**État** Terminé

**Terminé le** samedi 19 décembre 2020, 16:09

**Temps mis** 2 heures 29 min

**Points** 47,50/80,00

**Note** 5,94 sur 10,00 (59%)

Question 1

Correct

Note de 1,00 sur 1,00

Lequel de ces paradigmes ne fait pas partie des paradigmes de base de la cybersécurité :

- ☐ a. Cyber résilience
- ☐ b. Cyber défense
- ☐ c. Cyber protection
- ☒ d. Cyber détection



Votre réponse est correcte.

La réponse correcte est :

Cyber détection

Question 2

Correct

Note de 1,00 sur 1,00

Lequel de ces principes n'est pas un principe de base de la cyber résilience :

- ☐ a. L'adaptabilité
- ☒ b. La disponibilité
- ☐ c. L'absorbabilité
- ☐ d. La recouvrabilité



Votre réponse est correcte.

La réponse correcte est :

La disponibilité

## Question 3

Incorrect

Note de 0,00 sur 1,00

Je suis une technique de dissimulation de données dans des données, utilisée pour cacher des images, du texte et d'autres messages dans des images, des vidéos, de la musique ou des fichiers d'enregistrement. Je suis :

- ☐ a. La Cryptanalyse
- ☒ b. La Cryptographie
- ☐ c. La Tomographie
- ☐ d. La Stéganographie



Votre réponse est incorrecte.

La réponse correcte est :

La Stéganographie

## Question 4

Correct

Note de 1,00 sur 1,00

AES est l'acronyme de :

- ☐ a. Advanced Encrypted Standard
- ☐ b. Advanced Encryption Security
- ☐ c. Active Encryption Standard
- ☒ d. Advanced Encryption Standard



Votre réponse est correcte.

La réponse correcte est :

Advanced Encryption Standard

## Question 5

Correct

Note de 1,00 sur 1,00

Quel comportement malveillant consiste à remplir la boîte de courriel de la victime avec des courriers électroniques non sollicités ou indésirables ?

- ☐ a. Phishing
- ☐ b. Denial of Service
- ☒ c. Spamming
- ☐ d. Hooking



Votre réponse est correcte.

La réponse correcte est :  
Spamming

## Question 6

Correct

Note de 1,00 sur 1,00

Nous sommes des petits fichiers téléchargés dans votre système lorsque vous visitez un site web. Nous sommes les :

- ☐ a. Bots
- ☐ b. Crawlers
- ☒ c. Cookies
- ☐ d. Caches



Votre réponse est correcte.

La réponse correcte est :  
Cookies

## Question 7

Correct

Note de 1,00 sur 1,00

Dans un système de contrôle d'accès discrétionnaire :

- ☐ a. Les permissions d'accès données à un objet doivent rester cachées et à l'abri de regard « indiscrets » de potentiels attaquants
- ☐ b. Il est possible de définir des droits d'accès sur des groupes d'objets
- ☒ c. Seul l'administrateur peut changer le propriétaire d'un objet (« owner »), c'est-à-dire l'utilisateur à qui « appartient » un objet dans le système informatique ✓
- ☐ d. Seul le propriétaire d'un objet peut déterminer quels droits d'accès un utilisateur peut avoir sur cet objet

Votre réponse est correcte.

La réponse correcte est :

Seul l'administrateur peut changer le propriétaire d'un objet (« owner »), c'est-à-dire l'utilisateur à qui « appartient » un objet dans le système informatique

## Question 8

Incorrect

Note de 0,00 sur 1,00

L'utilisation d'une méthode d'authentification avec mot de passe à usage unique (« one-time password ») basée sur un secret partagé réduit le risque de compromission des comptes utilisateurs dans le cas où la base de données d'utilisateur est piratée.

Sélectionnez une réponse :

- ☐ Vrai
- ☒ Faux ✗

La réponse correcte est « Vrai ».

## Question 9

Correct

Note de 1,00 sur 1,00

Sous Linux, les applications s'exécutent généralement avec les droits d'accès de l'utilisateur qui a lancé l'application. Le fichier `/etc/shadow` peut seulement être modifié par l'utilisateur root. Comment est-ce que les utilisateurs peuvent changer leur propre mot de passe (contenu dans `/etc/shadow`) en utilisant le programme `passwd` ?

- ☒ a. Le programme `passwd` s'exécute avec les droits de root parce qu'il utilise le bit `setUID` ✔
- ☐ b. Les utilisateurs peuvent seulement changer leur mot de passe s'ils ont le mot de passe de root
- ☐ c. Le mot de passe de l'utilisateur est originellement dans `/etc/shadow`, et il est copié dans `/home/USER/shadow` où l'utilisateur peut le modifier.
- ☐ d. Le mot de passe de l'utilisateur est vérifié plus tard par un administrateur, qui fait la mise à jour de `/etc/shadow`

Votre réponse est correcte.

La réponse correcte est :

Le programme `passwd` s'exécute avec les droits de root parce qu'il utilise le bit `setUID`

## Question 10

Incorrect

Note de 0,00 sur 1,00

L'entropie peut être une mesure décrivant la difficulté de mener les attaques suivantes, à l'exception de :

- ☐ a. Une attaque de crackage de mot de passe par force brute.
- ☐ b. Une attaque de déni de service par SYN flooding.
- ☒ c. Une attaque de « session hijacking » dans une application Web utilisant des jetons de session (session ID). ✖
- ☐ d. Une attaque de cryptanalyse par analyse fréquentielle.

Votre réponse est incorrecte.

La réponse correcte est :

Une attaque de déni de service par SYN flooding.

## Question 11

Correct

Note de 1,00 sur 1,00

Une base de données de l'organisation X contenant des informations sensibles sur ses clients a fait l'objet d'une fuite et s'est retrouvée sur le marché noir, où elle a été vendue au crime organisé afin de leur permettre de commettre de la fraude. On soupçonne un employé de l'organisation d'avoir subtilisé ces informations, auxquelles il avait légitimement accès dans le cadre de ses fonctions, et de les avoir revendues. Quel facteur de l'analyse de risque est différent dans ce cas par rapport à un autre où un acteur externe aurait exploité une vulnérabilité des systèmes pour gagner accès à ces données, les copier et les revendre.

- ☐ a. Capacité
- ☐ b. Impact
- ☒ c. Opportunité
- ☐ d. Motivation



Votre réponse est correcte.

La réponse correcte est :

Opportunité

## Question 12

Incorrect

Note de 0,00 sur 1,00

On considère deux serveurs. Le premier combine un serveur web et un serveur FTP. Le second combine un serveur web et un serveur DNS. Chaque serveur est associé à une adresse privée dans le réseau local 192.168.0.1/24. On peut utiliser un unique pare-feu de type NetFilter pour filtrer les accès à ces deux serveurs.

Sélectionnez une réponse :

- ☒ Vrai ✖
- ☐ Faux


La réponse correcte est « Faux ».

## Question 13

Correct

Note de 1,00 sur 1,00

Laquelle de ces méthodes ne constitue pas une méthode de prévention des erreurs d'injection de code SQL

- ☒ a. La création d'un VPN SSL pour accéder à la base de données 
- ☐ b. L'utilisation d'un détecteur d'intrusion pouvant détecter les chaînes susceptibles d'être utilisées par une attaque d'injection de code SQL.
- ☐ c. L'utilisation de méthodes et fonctions directement implémentées sur le serveur de BD (« stored procedures »)
- ☐ d. L'utilisation de méthodes ou fonctions de filtrage des entrées venant des usagers

Votre réponse est correcte.

La réponse correcte est :


La création d'un VPN SSL pour accéder à la base de données

## Question 14

Correct

Note de 1,00 sur 1,00

Le principe qui dit que la sécurité d'un algorithme de cryptographie ne devrait dépendre que du secret de la clé

- ☐ a. A été énoncé par les inventeurs de l'algorithme RSA (Rivest, Shamir, Adleman)
- ☐ b. Ne s'applique qu'aux algorithmes de cryptographie à clé secrète
- ☐ c. N'est pas un principe de sécurité informatique
- ☒ d. S'appelle le principe de Kerchoff 

Votre réponse est correcte.

La réponse correcte est :

S'appelle le principe de Kerchoff

## Question 15

Partiellement correct

Note de 1,00 sur 2,00

Exploitation de vulnérabilité : Quelle est la différence entre une payload et un exploit ? (plusieurs réponses possibles)

- ☐ a. C'est la même chose
- ☒ b. L'exploit profite de la vulnérabilité pour faire exécuter la payload ✓
- ☒ c. La payload permet l'exécution de l'exploit ✗
- ☐ d. L'exploit correspond au lanceur d'une fusée alors que la payload c'est le satellite, ou encore l'exploit correspond au missile et la payload à la charge explosive

Votre réponse est partiellement correcte.

Vous en avez sélectionné correctement 1.

Les réponses correctes sont :

L'exploit correspond au lanceur d'une fusée alors que la payload c'est le satellite, ou encore l'exploit correspond au missile et la payload à la charge explosive,

L'exploit profite de la vulnérabilité pour faire exécuter la payload

## Question 16

Incorrect

Note de 0,00 sur 2,00

Laquelle de ces affirmations est vraie. Un pare-feu en mode personnel :

- ☐ a. Ne peut pas appliquer des règles de filtrage à états (« stateful » )
- ☒ b. Ne peut pas être installé sur une machine hôte qui possède une seule carte réseau ✗
- ☐ c. Ne peut filtrer que le trafic entrant sur la machine hôte
- ☐ d. N'a pas besoin de faire du NAT même si la machine hôte possède une adresse IP privée

Votre réponse est incorrecte.

La réponse correcte est :

N'a pas besoin de faire du NAT même si la machine hôte possède une adresse IP privée



## Question 17

Incorrect

Note de 0,00 sur 2,00

Le fichier examen.txt a été créé sur un système A en donnant les droits de lecture et écriture au propriétaire du fichier (rw-----). Le fichier est mis dans une clé USB et la clé est connectée dans un autre système B. Quel contrôleur de référence (système de contrôle d'accès) sera utilisé pour imposer les droits d'accès ?

- ☐ a. Le contrôleur de référence de la clé USB
- ☐ b. Le contrôleur de référence du système B
- ☒ c. Le contrôleur de référence du système A
- ☐ d. Aucune de ces réponses



Votre réponse est incorrecte.

La réponse correcte est :

Le contrôleur de référence du système B

## Question 18

Correct

Note de 2,00 sur 2,00

Donnez un exemple de défense dynamique et adaptative (Moving Target Defense)

- ☒ a. Address space layout randomization (ASLR)
- ☐ b. Canaries (StackGuard)
- ☐ c. StackShield
- ☐ d. Executable-space protection (ESP)



Votre réponse est correcte.

La réponse correcte est :

Address space layout randomization (ASLR)

## Question 19

Terminer

Note de 4,00 sur 4,00

Est-ce que le principe de segmentation d'un réseau et le principe de défense en profondeur sont reliés ? Si oui, expliquer comment ?

Je pense que le principe de segmentation d'un réseau et le principe de défense en profondeur est relié, car l'objectif de la segmentation d'un réseau est de diviser le réseau global en plusieurs sous-réseau ayant effet d'augmenter la performance et d'augmenter la sécurité puisqu'un attaquant, s'il pénètre un sous-réseau, il ne pourra pas se connecter aux autres sous-réseaux. La défense en profondeur consiste à ne pas faire confiance aux autres sous-systèmes et de se sécuriser soi-même en faisant abstraction du reste. Donc, si chaque sous-système se protège et assure sa sécurité, cela fait en sorte que si un attaquant pénètre un sous-système, il devra faire le même effort pour pénétrer les autres. ceci rends le système global plus sécuritaire. Donc, les deux sont reliés, car ils misent sur la division et le fait que si une partie est vulnérable, cela ne donne pas accès / permet pas l'attaquant d'accéder aux autres parties.

Les principes de segmentation d'un réseau et de défense en profondeur sont effectivement reliés dès lors que chaque segment du réseau est contrôlé par sa propre politique de filtrage. Sinon, cela ne sert à rien.

Si des pare-feu sont déployés pour filtrer l'accès à chaque segment, alors on atteint bien un objectif de défense en profondeur.

On peut notamment découper les segments en sous-segments pour renforcer la profondeur de la défense.

Commentaire :

## Question 20

Terminer

Note de 6,00 sur 6,00

Expliquez comment l'utilisation d'un mot de passe à usage unique (*One-Time Password ou OTP*) en combinaison avec des dispositifs d'authentification par identifiant et mot de passe permet de combiner les avantages des facteurs « quelque chose que je sais » avec « quelque chose que j'ai », par exemple dans le contexte d'application Web.

L'utilisation des OTP combine les deux avantages, car le mot de passe est quelque chose que l'utilisateur connaît, puisque c'est lui qui la définit. Le mot de passe unique est généralement envoyé par SMS au téléphone de l'utilisateur, donc, ceci est un objet que possède l'utilisateur. Dans le contexte d'applications web, utiliser de la biométrie statique ou dynamique est très difficile. Ceci donne tous les avantages d'une authentification à deux facteurs. En effet, l'attaquant, s'il connaît le mot de passe, ne pourra pas se connecter puisqu'il doit posséder aussi l'objet de l'utilisateur. De plus, avec les OTP, ceci élimine la possibilité des "replay attacks" puisque l'OTP est différent à chaque connexion.

Suite à l'authentification d'un usager U, avec son identifiant et son mot de passe, un OTP est généré sur le dispositif local avec une fonction à sens unique connue de tous (y compris Ève) à partir d'une chaîne codant l'intervalle de temps (timestamp) et un secret partagé S, associé à l'usager U. Cet OTP est alors copié par l'usager dans un troisième champ de saisie de données fourni à cet effet dans l'application Web. L'OTP reçu est vérifié par le serveur en recalculant la fonction avec le secret partagé S, obtenu de la base de données d'utilisateurs, et l'intervalle de temps actuel. Si le résultat est le même que l'OTP reçu, l'usager est authentifié.

Commentaire :

## Question 21

Correct

Note de 2,00 sur 2,00

Usurpation de certificat (3 sous-questions)

Bob.com est un serveur malveillant qui essaye de se faire passer pour le site légitime Charlie.com

Pour récupérer le certificat du serveur Charlie.com, il suffit que Bob.com fasse une demande de connexion HTTPS sur le site de Charlie.com

Sélectionnez une réponse :

- ☒ Vrai ✓
- ☐ Faux

La réponse correcte est « Vrai ».

Question **22**

Non répondue

Noté sur 2,00

## Usurpation de certificat (3 sous-questions)

Bob.com est un serveur malveillant qui essaye de se faire passer pour le site légitime Charlie.com

Lorsque le browser de Alice se connecte en HTTPS sur le site Web Bob.com, Bob.com lui présente le certificat valide du site Charlie.com. Dans ce cas, le browser d'Alice va vérifier le certificat et détecter que Bob.com a usurpé le certificat de Charlie.com. Le browser va rejeter le certificat et indiquer un message d'erreur à l'utilisateur.

Sélectionnez une réponse :

☐ Vrai☐ Faux

La réponse correcte est « Vrai ».

Question **23**

Non répondue

Noté sur 4,00

## Usurpation de certificat (3 sous-questions)

Bob.com est un serveur malveillant qui essaye de se faire passer pour le site légitime Charlie.com

Est-ce qu'au final Bob.com arrivera à établir une connexion HTTPS en utilisant le certificat du site Charlie.com ? Justifier la réponse.

La réponse est non sauf si Bob.com a volé la clé privée du serveur Charlie.com.

Lorsque le browser de Alice va vérifier la certificat envoyé par Bob.com, le browser de Alice va vérifier si ce certificat est valide : (1) il a été signé par une autorité de certification reconnue par Alice, (2) le hash associé au certificat confirme que le certificat est intègre, (3) Alice va demander à l'autorité de certification de confirmer que le certificat n'a pas été révoqué.

Ensuite, Alice va forger une clé de session et la transmettre en la chiffrant avec la clé publique du certificat (donc celle de Charlie.com).

Si Bob.com ne possède pas la clé privée de Charlie.com, il ne pourra pas déchiffrer le message envoyé par Alice et récupérer la clé de session forgée par Alice.

Bob.com ne pourra donc pas établir de session avec Alice en se faisant passer pour Charlie.com.

## Question 24

Correct

Note de 1,00 sur 1,00

Code vulnérable (5 sous-questions)

On considère le code suivant :

```
#include <stdio.h>
#include <stdlib.h>

void vuln(char *arg)
{
    int i=1,
    char buffer[4];
    strcpy(buffer, arg);
    if (i=0) printf("Cooooool !");
    if (i=1) printf(("Try again !"));
}

int main(int argc, char **argv)
{
    if (argc < 2) exit(0);
    vuln(argv[1]);
    exit(1);
}
```

Ce code est vulnérable à une attaque par débordement de pile (stack overflow)

Sélectionnez une réponse :

- ☒ Vrai ✓
- ☐ Faux

La réponse correcte est « Vrai ».

Question **25**

Correct

Note de 1,00 sur 1,00

Code vulnérable (5 sous-questions)

On considère le même code que celui de la question précédente :

```
#include <stdio.h>
#include <stdlib.h>

void vuln(char *arg)
{
    int i=1,
    char buffer[4];
    strcpy(buffer, arg);
    if (i=0) printf("Cooooool !");
    if (i=1) printf(("Try again !"));
}

int main(int argc, char **argv)
{
    if (argc < 2) exit(0);
    vuln(argv[1]);
    exit(1);
}
```

Le programme ci-dessus est appelé avec la chaîne de caractères « 123 ». La réponse du programme sera :

- ☐ a. Cooooool !
- ☒ b. Try again !



Votre réponse est correcte.

La réponse correcte est :

Try again !

## Question 26

Terminer

Note de 4,00 sur 4,00

Code vulnérable (5 sous-questions)

On considère le même code que celui de la question précédente :

```
#include <stdio.h>
#include <stdlib.h>

void vuln(char *arg)
{
    int i=1,
    char buffer[4];
    strcpy(buffer, arg);
    if (i=0) printf("Cooooool !");
    if (i=1) printf(("Try again !"));
}

int main(int argc, char **argv)
{
    if (argc < 2) exit(0);
    vuln(argv[1]);
    exit(1);
}
```

Le programme ci-dessus est appelé avec la chaîne de caractères « 1234 ». Expliquez pourquoi la réponse du programme sera « Cooooool ! »

Ici, la réponse sera cool, car la chaîne de caractère entrée par l'utilisateur est de longueur 5 au lieu de 4 puisqu'il faut ajouter le caractère de fin de chaîne qui est l'équivalent d'un caractère NULL. Puisque c'est 0 qui est utilisé pour indiquer la fin de la phrase, cette valeur va déborder et remplacer sur la pile la valeur du i.

Lorsque la fonction vuln est appelée, les variables int et char sont empilées sur la pile.

L'utilisateur va saisir la chaîne de caractère "1234" (de longueur 4).

Une fois cette chaîne saisie, celle-ci va être complétée par le symbole "00" pour indiquer la fin de la chaîne.

La chaîne "1234" va remplir le buffer char.

Le "00" va créer un débordement de buffer qui va écraser la variable i.

La variable i vaut donc 0 et donc le message "Try again !" est affiché.

Commentaire :

## Question 27

Terminer

Note de 2,00 sur 2,00

Code vulnérable (5 sous-questions)

On considère le même code que celui de la question précédente :

```
#include <stdio.h>
#include <stdlib.h>

void vuln(char *arg)
{ int i=1,
  char buffer[4];
  strcpy(buffer, arg);
  if (i=0) printf("Cooooool !");
  if (i=1) printf(("Try again !"));
}

int main(int argc, char **argv)
{
  if (argc < 2) exit(0);
  vuln(argv[1]);
  exit(1);
}
```

Le programme ci-dessus est appelé avec la chaîne de caractères « 12345 ». Quelle est la valeur de « i » dans ce cas ?

En hexadécimales, la valeur de i sera 0x0035. Il va contenir l'octet représentant le caractère '5' et le caractère de fin de chaînes.

Si le programme est appelé avec la chaîne de caractères « 12345 », alors c'est le caractère 5 qui va écraser la variable "i".

La valeur de "i" sera donc 5.

Commentaire :



## Question 28

Terminer

Note de 2,00 sur 2,00

Code vulnérable (5 sous-questions)

On considère le même code que celui de la question précédente :

```
#include <stdio.h>
#include <stdlib.h>

void vuln(char *arg)
{
    int i=1,
    char buffer[4];
    strcpy(buffer, arg);
    if (i=0) printf("Cooooool !");
    if (i=1) printf(("Try again !"));
}

int main(int argc, char **argv)
{
    if (argc < 2) exit(0);
    vuln(argv[1]);
    exit(1);
}
```

Le programme ci-dessus est appelé avec la chaîne de caractères « 123456789 ». Expliquer ce qu'il risque de se passer dans ce cas ?

Ici, nous avons rempli l'espace alloué pour les variables locales et nous avons débordé sur l'adresse de retour de la fonction. En effet, le caractère '9' et le caractère de fin de chaîne de caractère débordent sur l'espace alloué pour l'adresse de retour de la fonction. Ceci causera un "Segmentation Fault".

Dans ce cas, la chaîne de caractères "123456789" risque d'écraser la variable d'environnement ainsi que de l'adresse de retour. Cela risque donc de provoquer une erreur de type "segmentation fault".

Commentaire :

## Question 29

Terminer

Note de 1,00 sur 1,00

## La source des ennuis (8 sous questions)

On considère une source  $S_1$  markovienne qui génère des 0 et des 1. La probabilité d'apparition d'un 0 est de  $\frac{1}{4}$  et celle d'un 1 est de  $\frac{3}{4}$ . Quelle est l'entropie d'un message de 10 chiffres généré par la source  $S_1$  ?

$$H(s) = \text{somme } p_i \log_2 (1/p_i)$$

$$0.25 * \log_2(1 / (1/4)) = 0,5$$

$$0.75 * \log_2(1 / (3/4)) = 0,31$$

$$H(s) = 0,81 \text{ bits}$$

$$\text{Entropie de la source: } 10 * 0,81 = 8,1 \text{ bits}$$

On applique la formule pour calculer l'entropie de la source :

$$H(S_1) = 1/4 * \log_2(4) + 3/4 * \log_2(4/3)$$

$$= 1/2 + 0,311 = 0,811 \text{ bits}$$

Comme la source est markovienne (source aléatoire sans mémoire), il suffit de multiplier par 10 pour avoir l'entropie du message :

$$10 * 0,81 = 8,11 \text{ bits}$$

Commentaire :

## Question 30

Non répondue

Noté sur 2,00

La source des ennuis (8 sous questions)

En fait la source S1 contient un bug. Lorsque la source génère le premier chiffre, alors la probabilité d'apparition d'un 0 est bien de  $\frac{1}{4}$  et celle d'un 1 est de  $\frac{3}{4}$ . En revanche, pour le second chiffre, le résultat est le suivant :

- Si le premier chiffre est un 0, alors la probabilité d'apparition d'un 0 est de  $\frac{1}{2}$  et celle d'un 1 est aussi de  $\frac{1}{2}$ .
- Si le premier chiffre est un 1, alors la probabilité d'apparition d'un 0 est de  $\frac{1}{3}$  et celle d'un 1 est de  $\frac{2}{3}$ .

Le processus se répète de façon identique pour le troisième et quatrième chiffre : les probabilités associées au troisième chiffre sont identiques à celles du premier chiffre et les probabilités associées au quatrième chiffre sont identiques à celles du second chiffre.

Et ainsi de suite.

On appelle S2 cette seconde source.

Proposer une méthode M1 pour calculer l'entropie fréquentielle caractère par caractère de la source S2. Justifier votre réponse.

Soient  $P_i(0)$  et  $P_i(1)$ , la probabilité d'avoir respectivement un 0 ou 1 en position  $i$ .

On a donc  $P_1(0) = \frac{1}{4}$  et  $P_1(1) = \frac{3}{4}$

Soit  $P_2(0)$  et  $P_2(1)$ , la probabilité d'avoir respectivement un 0 ou 1 en deuxième position.

On a  $P_2(0) = P_2(0 | 0) + P_2(0 | 1) = \frac{1}{2} * \frac{1}{4} + \frac{1}{3} * \frac{3}{4} = \frac{3}{8}$

Et  $P_2(1) = P_2(1 | 0) + P_2(1 | 1) = \frac{1}{2} * \frac{1}{4} + \frac{2}{3} * \frac{3}{4} = \frac{5}{8}$

La fréquence d'apparition de 0 est égale à la limite quand  $n$  tend vers l'infini de  $\frac{\sum_{i=1}^n P_i(0)}{n}$

Comme le processus se répète respectivement sur les positions paires et les positions impaires, on a :

$P_{2n}(0) = P_2(0)$  et  $P_{(2n+1)}(0) = P_1(0)$

Si on note  $P_f(0)$  la probabilité fréquentielle de 0, on a donc  $P_f(0) = \frac{1}{2} (P_1(0) + P_2(0)) = \frac{1}{2} (\frac{1}{4} + \frac{3}{8}) = \frac{5}{16}$

On calcul de même  $P_f(1) = \frac{1}{2} (P_1(1) + P_2(1)) = \frac{1}{2} (\frac{3}{4} + \frac{5}{8}) = \frac{11}{16}$

L'entropie fréquentielle caractère par caractère de la source S2 sera donc égale à :

$H_f(S2) = P_f(0) * \log_2(1/P_f(0)) + P_f(1) * \log_2(1/P_f(1))$

Question **31**

Non répondue

Noté sur 2,00

La source des ennuis (8 sous questions)

Appliquer la méthode M1 pour calculer l'entropie caractère par caractère de la source S2. Soit E1 la valeur obtenue.

$$E1 = 5/16 * \log_2(16/5) + 11/16 * \log_2(16/11) = 0,3125 * 1,678 + 0,6875 * 0,540 = 0,896$$

Question **32**

Terminer

Note de 2,00 sur 2,00

La source des ennuis (8 sous questions)

On considère un message de longueur N générée par la source S2. Expliquer pourquoi l'entropie réelle de ce message n'est pas égale à  $N * E1$ . Cette entropie réelle est-elle supérieure ou inférieure à  $N * E1$  ?

Cette source semble avoir un peu de mémoire, puisque l'apparition du prochain caractère (pour le 2ème caractère et le quatrième ainsi de suite) dépend du résultat du premier. Donc, ici, il y a une "mémoire" qui est utilisée qui, dépendamment du caractère précédent, change les chances du prochain. Donc, ici, on pourrait coder par bloc. Donc, puisque les sources non markoviennes comme celle-ci ont une entropie plus petite ou égale à  $b * H(s)$ , l'entropie réelle du message n'est pas égale à  $N * E1$ , elle est inférieure.

La source S2 n'est pas markovienne car la probabilité d'apparition d'un caractère en position paire dépend de la probabilité d'apparition d'un caractère en position impaire.

L'entropie réelle d'un message générée par la source S2 n'est donc pas égale à  $N * E1$ .

Elle est strictement inférieure à  $N * E1$ .

Commentaire :

Question **33**

Terminer

Note de 2,00 sur 2,00

La source des ennuis (8 sous questions)

Proposer une méthode M2 pour calculer l'entropie réelle des messages générés par la source S2. Justifier votre réponse.

ici, on créerait un alphabet different  $S = \{00, 01, 11, 10\}$ . Pour savoir la probabilité de chacun, nous pouvons les combiner en multipliant les probabilités ensemble ce qui suit les règles des probabilités. En effet, lorsqu'on veut que deux événements arrivent en même temps, il faut multiplier les probabilités individuelles ensemble. Donc, pour la probabilité d'obtenir:

$$00 = 1/4 * 1/2 = 0,125$$

$$01 = 1/4 * 1/2 = 0,125$$

$$11 = 3/4 * 2/3 = 0,50$$

$$10 = 3/4 * 1/3 = 0,25$$

La somme de toutes les probabilités donnent 1.

Pour calculer l'entropie réelle de la source S2, il faut calculer l'entropie par digramme (bloc de deux caractères).

On considère donc le langage  $S2^2$  constitué de blocs de caractère en position impaire puis paire

Comme le processus de génération des caractères se répète tous les deux caractères, la source  $S2^2$  est donc markovienne.

Pour calculer l'entropie réelle Hr de la source S2, il faut donc calculer l'entropie de la source  $S2^2$  et on aura :  $Hr(S2) = H(S2^2) / 2$

Commentaire :

Question **34**

Terminer

Note de 1,00 sur 2,00

## La source des ennuis (8 sous questions)

Appliquer la méthode M2 pour calculer l'entropie d'un message de 10 chiffres généré par la source S2.

avec probabilité de 00 de 0.125, probabilité de 01 de 0.125, probabilité de 11 de 0,50 et probabilité de 10 de 0,25

pour 00:  $0.125 * \log(8) = 0,375$ pour 01:  $0.125 * \log(8) = 0,375$ pour 11:  $0.5 * \log(2) = 0,5$ pour 10:  $0.25 * \log(4) = 0,5$  $H(s) = 1,75$  bitsTotal:  $1,75 * 10 = 17,5$  bitsOn calcule l'entropie du langage  $S^2$ .

Il y a 4 digrammes possibles : 00, 01, 10, 11

On a  $P(00) = 1/4 * 1/2 = 1/8$ ,  $P(01) = 1/4 * 1/2 = 1/8$ ,  $P(10) = 3/4 * 1/3 = 1/4$ ,  $P(11) = 3/4 * 2/3 = 1/2$ Donc  $H(S^2/2) = 1/8 * \log_2(8) + 1/8 * \log_2(8) + 1/4 * \log_2(4) + 1/2 * \log_2(2) = 3/8 + 3/8 + 1/2 + 1/2 = 1,75$ 

Une chaîne de 10 caractères peut être divisée en 5 blocs de 2 caractères.

L'entropie d'une chaîne de 10 caractères générée par la source S2 sera donc égale à  $E_2 = 5 * 1,75 = 8,75$ On peut vérifier que 8,75 est bien inférieure à  $10 * E_1$ 

Commentaire :

Question **35**

Terminer

Note de 1,00 sur 2,00

La source des ennuis (8 sous questions)

Est-ce que la méthode M2 permet de calculer l'entropie du langage généré par la source S2 ? Justifier votre réponse

Non, car pour calculer l'entropie du langage, il faut évaluer la limite de  $H(s^b) / b$  avec  $b$  tendant vers l'infini. Donc, plus on utilise une valeur plus grande de  $b$ , plus on tend vers la limite ultime de compression. Nous avons mis  $b = 2$ . La méthode M2 ne représente pas cette limite ultime. Il faut vraiment calculer la limite pour savoir l'entropie du langage.

Comme le langage  $S2^2$  est une source markovienne aléatoire, la méthode M2 permet bien de calculer l'entropie du langage S2.

Commentaire :

Question **36**

Terminer

Note de 0,50 sur 2,00

La source des ennuis (8 sous questions)

On utilise les sources S1 et S2 pour générer des clés de longueur 128 bits. Est-ce que les clés générées par la source S1 sont plus faciles ou plus difficiles à casser que les clés générées par la source S2. Justifier votre réponse.

Ici, pour S1, le total avec 128 bits donnera: . Pour S2, celui-ci donnera: . Donc, on peut voir que l'entropie est plus grande pour

L'entropie d'une chaîne de 128 bits générée respectivement par les sources S1 et S2 sera de :

$128 * 0,811$  pour S1

$128 * 1,75 / 2$  pour S2

Les clés générées par S2 devraient être plus difficiles à casser que celles générées par S1.

Commentaire :



Question **37**

Terminer

Note de 4,00 sur 9,00

Configuration d'un pare-feu Netfilter (3 sous-questions)

On considère la configuration suivante d'un pare-feu Netfilter (il s'agit d'un extrait de la configuration vue en cours) :

```
# set default closed policy
iptables -P FORWARD DROP

# network interfaces
EXTIF=eth0
DMZIF=eth1

# addresses
EXTIP=195.55.55.1
WEB_SERVER=192.168.1.1
DNS_SERVER=192.168.1.2

# enable DNAT port translation from Internet to web server
iptables -t nat -A PREROUTING -i $EXTIF -p tcp --dport 80 -j DNAT --to-destination $WEB_SERVER:80

# enable DNAT port translation from Internet to dns server
iptables -t nat -A PREROUTING -i $EXTIF -p udp --dport 53 -j DNAT --to-destination $DNS_SERVER:53

# the web server must be accessible from Internet
iptables -A FORWARD -i $EXTIF -o $DMZIF -p tcp --dport 80 -m state --state NEW, ESTABLISHED -j ACCEPT

# the dns server must be accessible from Internet
iptables -A FORWARD -i $EXTIF -o $DMZIF -p udp --dport 53 -m state --state NEW, RELATED -j ACCEPT
```

On considère que les paquets suivants arrivent, dans cet ordre, sur les interfaces du pare-feu NetFilter (on suppose que la pare-feu n'a pas reçu de paquet avant Packet#1) :

Packet	Protocole	Src-IP	Dest-IP	Src-Port	Dest-Port	TCP Flags		
						SYN	SYN-ACK	ACK
Packet#1	TCP	195.5.5.1	192.168.1.1	2240	80	1	0	0
Packet#2	TCP	195.5.5.1	195.55.55.1	2240	80	1	0	0
Packet#3	TCP	195.5.5.1	195.55.55.1	2240	80	0	0	1
Packet#4	TCP	192.168.1.1	195.5.5.1	80	2240	0	1	0
Packet#5	TCP	192.168.1.1	195.5.5.1	80	2045	0	1	0

Packet	Protocole	Src-IP	Dest-IP	Src-Port	Dest-Port
Packet#6	UDP	195.5.5.1	195.55.55.1	3535	53
Packet#7	UDP	192.168.1.2	195.5.5.1	53	3535
Packet#8	UDP	192.168.1.2	195.4.4.1	53	3535

Packet	Protocole	Src-IP	Dest-IP	TYPE	CODE	Payload attributes

						Src-IP	Dest-IP	Src-Port	Dest-Port
Packet#9	ICMP	192.168.1.2	195.5.5.1	3	3	195.5.5.1	195.55.55.1	3535	53

Les paquets Packet#1, Packet#2, Packet#3 et Packet#6 arrivent sur l'interface EXTIF du pare-feu.

Les paquets Packet#4, Packet#5, Packet#7, Packet#8 et Packet#9 arrivent sur l'interface DMZIF du pare-feu.

Pour chaque paquet de Packet#1 à Packet#9, indiquer si le paquet sera accepté ou bloqué par le pare-feu Netfilter. Justifier la réponse.

Les paquets acceptés sont les paquets #1 et #4. Le #1 sera accepté puisque les règles permettent le passage de eth0 vers WEB Server. Pour le paquet #4, il le sera aussi, car il suit les mêmes configurations que le paquet #1. Puisqu'il a déjà accepté cela, c'est une connexion établie. Donc, il le laissera passer.

Les paquets #2 et #3 seront refusés puisqu'il n'existe pas de règles vers DNS\_SERVER.

Les paquets #6, #7 et #8 seront bloqués, puisqu'il n'existe pas de paquets ayant établie une connexion avec le port 3535.

Le paquet 5 sera bloqué, car il n'y a pas d'autres paquets qui ont établi une connexion avec le port 2045.

Le paquet 9 sera bloqué, car il n'y a pas de protocole ICMP défini. Puisque pour FORWARD c'est drop par défaut, il sera bloqué.

Packet#1 : refusé car paquet non routable à cause de l'adresse privée.

Packet#2 : accepté et redirigé vers le serveur web après NAT

Packet#3 : refusé car paquet ACK hors connexion (le serveur n'a pas encore envoyé le SYN-ACK)

Packet#4 : accepté car réponse du serveur au paquet SYN. Sera envoyé à l'adresse 195.5.5.1 après NAT.

Packet#5 : refusé car paquet hors session (port destination incorrect)

Packet#6 : accepté et redirigé vers le serveur DNS après NAT

Packet#7 : accepté car réponse du serveur DNS traité comme trafic RELATED

Packet#8 : refusé car réponse hors session (adresse destination incorrecte)

Packet#9 : accepté car le pare-feu va considérer qu'il s'agit d'un message d'erreur envoyé par le serveur DNS en réponse à la demande du client

Commentaire :

Question **38**

Terminer

Note de 1,00 sur 3,00

## Configuration d'un pare-feu Netfilter (3 sous-questions)

Pour le Packet#9, expliquer ce qu'il peut se passer au niveau du pare-feu et à quel type de comportement malveillant cela peut correspondre.

Il se peut que si l'on envoie un paquet UDP sur le port 53 et que la destination n'est pas un serveur DNS, puisque la destination va renvoyer un message ICMP port unreachable, tout paquet va

La pare-feu va accepter la paquet en considérant qu'il s'agit d'un message d'erreur envoyé par le serveur DNS.

Un attaquant peut utiliser cette possibilité pour générer du trafic ICMP en spoofant l'adresse du serveur DNS.

Cela peut mettre le pare-feu en déni de service (DOF - Denial of Firewall).

Ce comportement malveillant correspond à l'attaque Black-Nurse (voir dernier acétate du cours de Sécurité Réseau 1).

Commentaire :

Question **39**

Non répondue

Noté sur 3,00

## Configuration d'un pare-feu Netfilter (3 sous-questions)

Lorsque les 9 paquets ci-dessus sont reçus par le pare-feu, combien de sessions sont présentes dans la table de session interne du pare-feu ? Justifier la réponse en indiquant le nombre de sessions TCP, UDP et ICMP présentes dans la table de session (on suppose qu'aucun des timeouts associés aux sessions n'est atteint).

Il y aura deux sessions dans la table de session :

- Une session TCP entre 155.5.1.1 et 192.168.1.1 dans l'état NEW
- Une session UDP entre 155.5.1.1 et 192.168.1.2

◀ Vidéo Métiers et Gestion de la Sécurité Partie 3

Aller à...