ÉCOLE
**POLYTECHNIQUE**
MONTRÉAL

# INF4420A – Sécurité Informatique

## Travail Pratique 1

15-09-09

# Sommaire

## Directives

Tous les travaux devront être remis avant 23h59 le jour de la remise sur le site Moodle du cours. À moins que cela ne soit explicitement demandé dans le sujet, vous ne devez remettre qu'un fichier PDF nommé selon le format *TPX-matricule1-matricule2.pdf*. Vous pouvez inclure des annexes dans votre rapport si vous jugez que cela améliore la lisibilité (code source, …)

- Voir la date de remise du rapport de ce laboratoire dans le plan du cours.
- Le travail devra être fait par équipe de deux. Toute exception (travail individuel, équipe de trois) devra être approuvée au préalable par le professeur.
- L'orthographe et la forme seront prises en compte pour chaque question.
- Indiquez toutes vos sources d'information, qu'elles soient humaines ou documentaires.

**NOTE : POUR TOUTES LES QUESTIONS, VOUS DEVREZ MONTRER COMMENT VOUS AVEZ OBTENU LES RÉPONSES EN REPRODUISANT DANS VOTRE RAPPORT LES COMMANDES UTILISÉES ET LEUR SORTIE.**

# Partie A

## Sources d'information

Il a été vu en classe que la notion abstraite d'une source peut être décrite comme une boîte noire qui, à chaque fois qu'on lui demande (qu'on pèse sur le « bouton »), donne un nouveau symbole. L'ensemble de symboles qu'une source peut générer est son "alphabet". Pour les besoins de ce travail pratique, nous allons « simuler » ces boîtes abstraites par des programmes exécutables. Ces programmes se trouvent dans l'archive « Utilitaires TP1 » sur le site Moodle :

- `monnaie` génère des 0 et des 1 de façon indépendante avec la même probabilité (1/2).
- `binaire` génère des 0 et des 1 selon un algorithme « secret ».
- `lettre` génère une lettre de façon aléatoire. La probabilité qu'une lettre déterminée soit générée correspond à sa fréquence dans la langue anglaise.
- `texte` est similaire à `lettre`, excepté qu'à chaque invocation le nombre de lettres indiqué sont tirées, dans l'ordre, d'un vrai texte en anglais. À chaque invocation, le texte choisi est différent.

Dans les deux derniers cas, il s'agit de lettres de l'alphabet anglais (26 lettres) et l'espace. Dans le cas de `texte`, l'apostrophe est représentée par un espace.

L'usage de ces programmes est très simple : à chaque invocation du programme, un nouveau symbole est généré et affiché à la console :

```
$ ./lettre
E
```

Vous pouvez spécifier un argument numérique qui spécifie le `nombre de symboles qui seront générés :

```
$ ./lettre 4
E GK
```

Vous pouvez utiliser les commandes de redirection « > », « < » pour rediriger la sortie de la source vers un fichier, ou même « | » pour passer la sortie par un autre programme.

Les compromis d'implémentation (« features ») suivants ont été introduits:

- Pour les sources « binaires » (`monnaie` et `binaire`), les bits sont générés par blocs de 8 et l'argument optionnel indique donc le nombre **d'octets** générés.
- Pour les sources « alphabétiques » (`lettre` et `texte`), ce qui est généré est les versions ASCII des lettres majuscules correspondantes.

La visualisation directe de la sortie des sources alphabétiques est possible sur la console ou via un utilitaire comme `cat` ou `more`. Quoique la sortie des sources binaires puisse être observée directement, il est préférable d'utiliser un éditeur ou un visualisateur de fichiers binaires, par exemple `fb` (gratuit) ou celui de Visual C++.

## Entropie

Nous vous avons fourni trois outils qui vous permettront d'estimer l'entropie des différentes sources. Ils calculent tous l'entropie de l'entrée, mais ce de façon différente :

- `h-bit` calcule les fréquences bit par bit, et calcule l'entropie « par bit », c'est-à-dire comme si l'entrée était la sortie d'une source binaire. L'utilitaire utilise l'entrée standard. Par exemple,

```
$ ./monnaie 128 > monnaie.bin
$ ./h-bit < monnaie.bin
0 = 490
1 = 534
Nombre total de bits : 1024
Entropie du texte entre : 0.998668
```

Notez que dans notre cas, et étant donné que la sortie des sources « alphabétiques » est déjà codée en binaire (via le codage ASCII) nous pouvons aussi appliquer cet outil sur ces sources.

```
$ ./lettre 128 > texte.bin
$ ./h-bit < texte.bin
0 = 647
1 = 377
Nombre total de bits : 1024
Entropie du texte entre : 0.949252
```

- `h-ascii` calcule les fréquences octet par octet d'une source, et calcule l'entropie « par octet », c'est-à-dire en considérant l'entropie comme s'il s'agissait d'une source générant des octets. Tout comme h-bit, l'utilitaire utilise l'entrée standard.

- `h-lettre` calcule les fréquences octet par octet, mais seulement pour les codes ASCII représentant des lettres majuscules et l'espace. À partir de ces fréquences, l'entropie (en bits) est calculée pour une source générant des lettres majuscules et des espaces. Cet utilitaire utilise aussi l'entrée standard.

## Chiffrement

Nous avons fourni quelques programmes de chiffrement simples:

- `cesar` et `cesar-d` sont des programmes pour chiffrer (`cesar`) et déchiffrer (`cesar-d`) selon la méthode de Jules César. Ils fonctionnent correctement seulement sur les codes ASCII représentant des lettres majuscules et sur les espaces (ces derniers sont préservés). Par exemple,

```
$ ./cesar
ALEA JACTA EST
DOHD MDFWD HVW
$ ./cesar-d
DOHD MDFWD HVW
ALEA JACTA EST
```

- `masque` est une implémentation du masque jetable (« one-time pad »). L'utilitaire nécessite 4 arguments qui sont les suivants, en ordre :
    - fichier contenant la clé (qui doit se trouver dans le même répertoire que le programme)
    - taille de la clé (en octets)
    - fichier contenant le message à chiffrer (qui doit se trouver dans le même répertoire que le programme)
    - fichier de sortie qui contiendra le message chiffré (qui sera créé dans le même répertoire que le programme)

## Question 1 - Entropie [/0.75]

a) Calculez l'entropie par lettre (`h-lettre`) d'une chaîne générée avec `texte` d'une longueur de 200 caractères.

b) En vous servant du premier théorème de Shannon, expliquez ce que signifie cette valeur.

c) Quelle serait l'entropie par lettre (en moyenne) d'un fichier qui aurait été généré de la même façon, mais avec les mêmes probabilités (1/27) pour chacun des 27 symboles (lettres majuscules et espace)?

d) Que représente le quotient de la valeur en a) sur la valeur en c) ?

e) Refaites la même chose qu'en a) avec la source `lettre`. Comparez la valeur obtenue avec celle en a). Est-ce que la différence est significative (supérieure à 0.4) ?

f) On sait qu'un texte anglais est constitué de mots et de phrases qu'il est nécessaire d'interpréter en fonction d'un langage et d'une grammaire. Un texte anglais est donc très redondant (et donc facile à comprimer). Les chaînes générées par `lettre` ne sont pas de l'anglais malgré l'utilisation des mêmes fréquences. Le résultat obtenu en e) peut donc surprendre. Expliquez cette contradiction apparente (le fait que les deux entropies soient proches).

## Question 2 - Histogrammes [/0.75]

a) Utilisez les programmes `cesar` et `cesar-d` avec les sources `texte` et `lettre`, pour chiffrer et déchiffrer des chaînes de 200 caractères.

b) Utilisez le programme `h-lettre` pour obtenir les fréquences des lettres. Construisez des histogrammes de fréquences ordonnées du plus grand au plus petit pour la sortie de chacune des sources ainsi que pour les versions codées. (Note : Vous pouvez facilement générer ces histogrammes en redirigeant la sortie de `h-lettre` dans un fichier que vous pouvez importer et traiter dans Excel, par exemple).

c) Que remarquez-vous en comparant ces quatre histogrammes? Comment seraient les histogrammes des sources `lettre` et `texte` si les fréquences étaient comptabilisées sur deux lettres à la fois? Comment devrait être par exemple les fréquences du (ee) et du (th) dans le cas de `texte` et de `lettre`.

d) En vous référant au point précédent ainsi qu'à la question 1 f), est-ce que cette méthode (comptabiliser les fréquences sur deux lettres) facilite le déchiffrement du message dans le cas de la source `texte` ? Et dans le cas de `lettre` ? Expliquez la différence s'il y en a une. Pour chacune des deux sources, si cette méthode n'augmente pas la facilité de déchiffrement du message, quelle solution proposez-vous ?

## Question 3 - Masque jetable [/0.75]

a) Générez un fichier de 1024 octets avec `monnaie` et un avec `binaire`. Calculer l'entropie par bit (`h-bit`) et l'entropie par octet (`h-ascii`) sur les deux fichiers créés.

b) Générez une clé de 1024 octets pour un masque jetable avec `monnaie` (tel que vu en classe, la taille de la clé doit être la même que la taille du message à chiffrer). Appliquez le masque jetable sur les deux fichiers générés au point précédent en utilisant la clé nouvellement créée. Calculer l'entropie par bit (`h-bit`) et l'entropie par octet (`h-ascii`) des nouveaux fichiers chiffrés. Qu'observez-vous? Quelles conclusions pouvez-vous en tirer?

c) Pour les deux cas, s'agit-il d'une méthode sécuritaire de chiffrement?

## Question 4 - Analyse de risque [/1.5]

La compagnie PokerMaxProUltime, une nouvelle compagnie de poker sur Internet, vous offre un emploi de rêve en tant qu'analyste principal de sécurité pour leur nouvelle installation dans un paradis fiscal aux Caraïbes. Vous devez toutefois faire face à plusieurs défis.

a) Pour commencer, votre patron vous indique que deux sites potentiels sont retenus pour la nouvelle installation. Le site A se situe sur une île paisible, mais où le marché immobilier est gonflé par les étrangers. Il en coûterait donc 500 000 $ pour s'installer à cet endroit. L'île B, pour attirer des capitaux étrangers, a fait une proposition à votre compagnie. Il en coûterait seulement 100 000 $ pour s'installer sur le site B. Toutefois, selon les données météos que vous avez à votre disposition, l'île B est balayée chaque année par un ouragan qui a 25% de chance de détruire votre installation. Quelle serait votre recommandation et pourquoi ?

b) Vous rencontrez les gestionnaires des diverses lignes d'affaires, et vous évaluez leurs processus d'affaire pour identifier les risques. Trois risques majeurs en ressortent :

   i. Un malfaiteur ignore les lignes de conduite prescrites (*Terms of Service*) et utilise les fonctions du logiciel pour tricher, diminuant l'intérêt du site pour les joueurs légitimes.
   ii. Un malfaiteur inonde le serveur de requête pour empêcher les autres joueurs de se connecter à votre site.
   iii. Un malfaiteur infiltre votre base de données pour obtenir certaines des informations que vous stockez sur vos clients (adresses courriel et postal, numéro de carte de crédit, habitudes de jeu, historique des achats).

   Pour chacun de ces scénarios, précisez s'il s'agit principalement d'un scénario touchant l'intégrité, la confidentialité ou la disponibilité.

c) Après de longs mois d'études, vous avez identifié trois agents de menace potentiels pour votre entreprise :

- Tricheurs professionnels : gens qui s'y connaissent peu en informatique, mais beaucoup au jeu;
- Crime organisé : groupes criminalisés qui ont plusieurs experts à leur solde et qui possèdent une solide infrastructure avec des milliers d'ordinateurs compromis;
- Sites de poker concurrents : le jeu en ligne est un milieu lucratif et certains de vos concurrents sont prêts à tout pour connaître le secret de votre succès.

Votre patron vous fournit le résultat de l'étude de risque qu'il a fait faire par un grand cabinet de conseil et vous demande de le compléter :

| | Acteur | Capacité | Opportunité | Motivation | Probabilité | Impact | Risque |
|---|---|---|---|---|---|---|---|
| Scénario i | Tricheur | 4 | 4 | 4 | | 2 | |
| | C.O. | 1 | 4 | 1 | | 2 | |
| | Concurrents | 2 | 4 | 2 | | 2 | |

| | Acteur | Capacité | Opportunité | Motivation | Probabilité | Impact | Risque |
|---|---|---|---|---|---|---|---|
| Scénario ii | Tricheur | 1 | 4 | 1 | | 4 | |
| | C.O. | 4 | 4 | 1 | | 4 | |
| | Concurrents | 2 | 4 | 4 | | 4 | |

| | Acteur | Capacité | Opportunité | Motivation | Probabilité | Impact | Risque |
|---|---|---|---|---|---|---|---|
| Scénario iii | Tricheur | 1 | 3 | 1 | | 3 | |
| | C.O. | 4 | 3 | 4 | | 3 | |
| | Concurrents | 1 | 3 | 2 | | 3 | |

Échelle :

| | |
|---|---|
| 1 | Très peu ou nul |
| 2 | Peu |
| 3 | Élevé |
| 4 | Très élevé |

Commentez, pour chaque scénario de risque, quel serait l'acteur qui constitue la plus grande menace pour votre entreprise.

d) Pour chacune des situations suivantes expliquez quel(s) paramètre(s) changerai(en)t et dans quel sens (plus grand, plus petit). Quelle(s) conséquence(s) pour la gestion du risque ?
    1. Votre compagnie de poker remporte un très grand succès et dépasse tous vos concurrents.
    2. Votre patron a refusé de payer les pots-de-vin réclamés par la mafia locale
    3. Votre patron fait l'acquisition d'un tout nouveau système de détection des tricheurs très performant.

e) Un vendeur vous propose un service de surveillance à distance pour faire de la détection d'intrusion sur vos serveurs. Il suffit d'installer un logiciel de surveillance et de contrôle à distance pour permettre à ce fournisseur de détecter et combattre les intrusions. Celui-ci vous offre le service à très bon marché (5 000 $ par mois pour une surveillance 24h sur 24) puisqu'il vient d'ouvrir un nouveau centre d'opération dans un pays de l'ex-Union Soviétique où la main d'œuvre coûte une fraction de la main d'œuvre au Canada. Refaites la grille de la question c) pour le scénario iii) en prenant en compte la mesure proposée. Est-ce que vous croyez que cette offre en vaut la chandelle ? Est-ce que votre recommandation s'applique dans toutes les circonstances ?

# Partie B

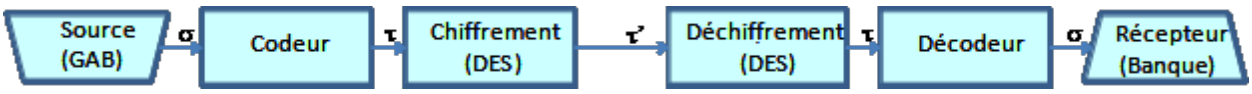
## Question 1 - Codage [/1.25]

* Les fichiers nécessaires se trouvent dans le dossier « Codage » de l'archive « Utilitaires TP1 » sur le site Moodle.

La Banque de Chibougamau a installé dans chacune de ses succursales une machine d'écriture de carte bancaire qui permet au client de changer le NIP (numéro d'identification personnel de 4 chiffres) associé à leurs cartes bancaires et qui est utilisé comme mot de passe par les guichets automatiques bancaires (GAB). Comme les NIPs doivent être envoyés à l'ordinateur central de la banque pour que les GAB puissent les vérifier lors de transactions bancaires, les machines d'écriture enregistrent les nouveaux NIP et les envoient à l'ordinateur central par réseau à chaque fois qu'un NIP est changé.

Actuellement, le guichet automatique encode un NIP en prenant la suite de 4 chiffres encodés en ASCII répété 2 fois pour détecter les erreurs de transmission. Donc, le NIP « 1,2,3,4 » serait représenté par la séquence « 12341234 » en ASCII. Ceci créé donc un bloc de 8 octets, donc 64 bits. Pour éviter que les NIPs soient interceptés, ce bloc est alors chiffré avec l'algorithme DES avant d'être envoyé par réseau. Puisqu'on envoie qu'un bloc à la fois, il n'y a pas de chainage. Aussi, étant donné que la clé de chiffrement est stockée dans une puce à l'épreuve des manipulations à l'intérieur du guichet automatique, il est assez difficile de changer la clé.

Avec l'implémentation décrite, le système de transmission des NIPs est possiblement vulnérable. Vous devez développer un codeur pour remplacer le codage existant afin de minimiser la vulnérabilité du système à l'interception. Le reste de la transmission (identité du client, …) est considérée sûre et indépendante.



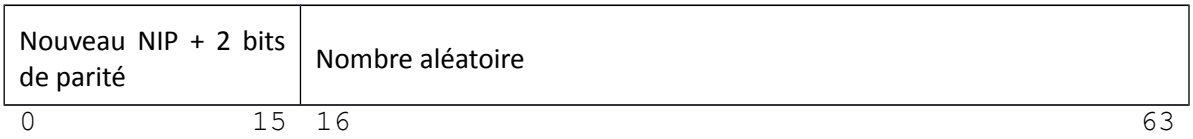

**Cas où le codage est inchangé :**

a) Expliciter les alphabets σ, τ et τ' qui sont respectivement les alphabets pour la sortie de la source, du codeur et du bloc de chiffrement.
b) Identifiez les langages provenant des alphabets σ, τ et τ'
c) Ensuite, identifiez les attaques auxquelles le système est vulnérable. Pour identifier ces attaques, rappelez-vous qu'un attaquant peut connaitre parfaitement le fonctionnement des boîtes de codage et chiffrement mais qu'il n'a bien sûr pas accès à la clé. Aussi, un attaquant peut intercepter tous les messages chiffrés et même les modifier.
d) Pour chacune des attaques identifiées au c), montrez à l'aide de traces d'exécution comment vous les effectueriez. Pour cela, utilisez les scripts `transBase` et `recepBase` qui implémentent respectivement les blocs *source+codeur+chiffrement* et *déchiffrement+décodeur+récepteur.*

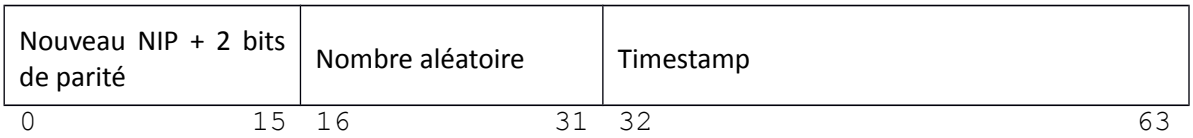
**Cas où l'on change de codage :**

Trois nouveaux codeurs vous sont proposés :

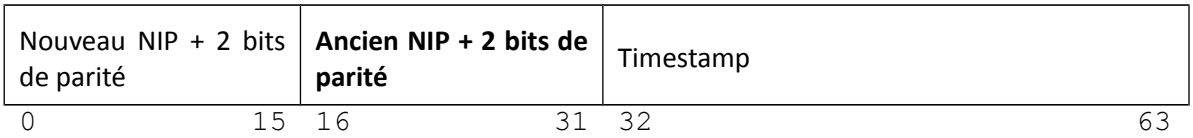
- Codage1 : le nouveau NIP est codé en binaire sur 14 bits plus 2 bits de parité (les détails ne sont pas importants). Un nombre aléatoire sur 48 bits est concaténé pour former un bloc de 64 bits.
  Détails :




- Codage2 : le nouveau NIP est codé en binaire sur 14 bits plus 2 bits de parité. Un nombre aléatoire de 16 bits puis un « timestamp » Unix de 32 bits sont concaténés pour former un bloc de 64 bits.
  Détails :




- Codage3 : le nouveau et l'ancien NIP sont codés en binaire sur 14 bits plus 2 bits de parité puis concaténés. Un « timestamp » Unix de 32 bits est concaténé pour former un bloc de 64 bits.
  Détails :




Remarque : on suppose que le « timestamp » peut être utilisé par la banque pour invalider les messages dont le timestamp est trop vieux.

Les binaires `trans1`, `recep1`, `trans2`, `recep2`, `trans3` et `recep3` sont les équivalents de `transAscii` et `recepAscii` mais avec les codages correspondants.

e) Pour chacun des trois codages, dites quelles attaques du c) ils permettent de bloquer et démontrez-le à l'aide de trace d'exécution.

f) Selon vous quel est le meilleur codage ? Pourquoi ?

## Question 2 - Certificats à clé publique, HTTPS et SSL [/1]

Pour faire de l'hameçonnage, les pirates créent un site identique à un site d'une institution bancaire pour faire croire à la victime qu'elle se trouve sur le site véritable. Lorsque la victime entre ses informations confidentielles, son numéro de compte et son mot de passe par exemple, pour effectuer ses transactions, un message d'erreur tel que « le serveur est plein et ne peut traiter la requête » est retourné à la victime et ses informations confidentielles sont envoyées aux pirates. Pour remédier (en partie) au problème, les certificats sont utilisés avec les connexions HTTP sécurisées utilisant le protocole SSL (HTTPS). Pour la prochaine question, vous devez démarrer la machine virtuelle nommée Certificates dans VMWare Workstation. Le mot de passe de l'utilisateur admin_web est « **SeCsIlab** ».

a) Essayez de vous connecter au faux site de la Caisse Desjardins à l'adresse https://www.desjardins.com à l'aide de Firefox. Que se passe-t-il et pourquoi?

b) Qu'est-ce qui pourrait vous aider à découvrir que le site est une fraude?

c) Ajoutez l'exception de sécurité de façon permanente, c'est-à-dire le certificat « self-signed ». Redémarrez Firefox et réessayez de vous connecter à l'adresse https://www.desjardins.com. Quel est maintenant le nouveau comportement de Firefox et pourquoi?

d) Effacez le certificat que vous venez d'ajouter dans Firefox en suivant les instructions suivantes :

1. Démarrez Firefox
2. Allez dans le menu Edit -> Preferences
3. Sous l'onglet Advanced, ouvrez l'onglet Encryption
4. Cliquez sur View Certificates
5. Le certificat ajouté se trouve sous l'onglet Servers. Si vous n'avez pas regardé le certificat (ce que vous auriez dû faire lorsque vous l'avez ajouté), vous pouvez le faire en le sélectionnant et en appuyant sur le bouton View
6. Pour l'effacer, sélectionnez-le et appuyez sur le bouton Delete

Pour vous assurer que le certificat est bien effacé, essayez de vous reconnecter à l'adresse https://www.desjardins.com et d'actualiser la page. Vous devriez recevoir le même message qu'à la question a). **N'ajoutez pas l'exception**.

Une fois le certificat effacé, vous allez générer une clé puis un certificat CA (Certificate Authority) qui permet de signer d'autres certificats :

```
openssl genrsa -des3 -out ca.key 4096
```

Mémoriser la « pass phrase »…

```
openssl req -new -x509 -days 365 -key ca.key -out ca.crt
```

Répondez aux questions comme bon vous semble.

Le certificat créé est `ca.crt` et se trouve dans le répertoire courant (surement ~/). Ajoutez le certificat dans Firefox en suivant la procédure suivante:

1. Démarrez Firefox
2. Allez dans le menu Edit -> Preferences
3. Sous l'onglet Advanced, ouvrez l'onglet Encryption
4. Cliquez sur View Certificates
5. Sous l'onglet Authorities, appuyez sur le bouton Import
6. Choisissez le certificat CA que vous venez de créer
7. Cochez toutes les cases et appuyez sur le bouton OK. Vous pouvez visualiser votre certificat en appuyant sur le bouton View

Vous allez maintenant générer un nouveau certificat pour https://www.desjardins.com puis le signer avec votre certificat CA. Pour ce faire, terminez d'abord le serveur web grâce à la commande suivante:

```
sudo /etc/init.d/apache2 stop
```

Génération d'une clé puis d'une requête de signature pour www.desjardins.com :

```
openssl genrsa -des3 -out desjardins.key 4096
```

Mémoriser la « pass phrase »…

```
openssl req -new -key desjardins.key -out desjardins.csr
```

Répondez aux questions comme bon vous semble SAUF pour « Common Name » où vous devez répondre www.desjardins.com. Ne donnez pas de « challenge password ».

Signature de la requête :

```
openssl x509 -req -days 365 -in desjardins.csr -CA ca.crt -CAkey
ca.key -set_serial 01 -out desjardins.crt
```

Afin d'éviter qu'Apache demande la « pass phrase » du certificat à chaque démarrage, générer une version non sécurisée de la clé desjardins`.key` :

```
openssl rsa -in desjardins.key -out desjardins.key.insecure
mv desjardins.key desjardins.key.secure
mv desjardins.key.insecure desjardins.key
```

Copiez maintenant le certificat signé et sa clé non sécurisée dans le répertoire d'Apache :

```
sudo cp desjardins.crt desjardins.key /etc/ssl/apache2/
```

Finalement, démarrez le serveur web avec la commande suivante:

```
sudo /etc/init.d/apache2 start
```

Ouvrez maintenant Firefox et essayez de vous connecter à l'adresse https://www.desjardins.com. Quel est le comportement de Firefox et pourquoi?

e) Avec Firefox, essayez de vous connecter aux sites https://www.rbc.com et https://www.bmo.com. Que se passe-t-il ?

f) Sur le site www.bmo.com ajoutez une exception de sécurité temporaire (ne pas cocher « Permanently store this exception »). Accédez au site puis changer de site (allez sur about:home par exemple). Effacez le cache de Firefox :

1. Appuyez sur Ctrl+Shift+Suppr
2. Sélectionnez « Everything » dans « Time range to clear »
3. Dans « Détails » cochez toutes les cases
4. Cliquez sur « Clear Now »

Retournez sur le site www.bmo.com et expliquez pourquoi vous n'avez plus accès au site.

g) Allez sur le site www.rbc.com et ajouter une exception de sécurité temporaire. Que se passe-t-il ? Cochez les trois cases et validez. Changez de site et effacez le cache de Firefox comme au f). Retournez sur www.rbc.com. Que se passe-t-il ? Expliquez.

h) Aller sur www.bmo.com. Que se passe-t-il ? Expliquez.

i) À la lumière des résultats que vous avez obtenus au long de tout cet exercice, pourquoi est-il dangereux d'accepter des certificats « self-signed » et, pire encore, des certificats CA?

## Question 3 - Chiffrement par bloc et modes d'opération [/0.5]

L'administrateur réseau de la compagnie WebFacile se soucie des problèmes de vol de mot de passe par des malwares « sniffant » les fichiers textes. N'arrivant pas à se rappeler de tous les mots de passe de son réseau il décide de les enregistrer sous forme d'image puis de les chiffrer. Il décide d'utiliser un chiffrement AES 256 bits car il sait que celui-ci est sécuritaire. Cependant, ne comprenant pas bien à quoi correspondent les différents modes d'opération il utilise le premier : ECB (Electronic Code Book).

* Les fichiers nécessaires se trouvent dans le dossier « Chiffrement par bloc » de l'archive « Utilitaires TP1 » sur le site Moodle.

a) Le fichier mdp.jpg est un des mots de passe de l'administrateur enregistré sous forme d'image. À l'aide du script python AES.py, chiffrez ce fichier en mode ECB. (Exécutez le script sans argument pour connaître son fonctionnement). Observez le fichier de sortie et commentez.

b) Chiffrez maintenant le fichier en mode CBC. Observez le fichier puis commentez.

c) Concluez sur l'importance des modes d'opération des algorithmes de chiffrement par bloc.

## Question 4 - Organisation des mots de passe en UNIX/Linux [/1]

Le but de cette question est de vous familiariser avec les mécanismes de protection de mot de passe en Linux. Vous allez utiliser la machine virtuelle **Mdp** dont le mot de passe de `root` est **`Af6Thg+9`**.

**a)** Examinez le fichier `/etc/passwd`. Contient-il des mots de passe ? Pourquoi? Quelles sont ses permissions d'accès? Pourquoi ?

**b)** Observez les fichiers `passwd` et `shadow` qui se trouvent sous le répertoire `/etc/`.
Ajoutez un utilisateur avec la commande:
```
$ useradd -g users -s/bin/bash -m NOM
```
avec `NOM`= le nom de l'utilisateur que vous ajoutez
Donnez un password à l'utilisateur que vous avez créé avec la commande:
```
$ passwd NOM
```
Observez ce qui se passe dans les fichiers `passwd` et `shadow`. Lequel ou lesquels de ces deux fichiers sont modifiés ? Pourquoi ?

**c)** Changez le mot de passe de l'utilisateur que vous venez de créer avec la commande
```
$ passwd NOM
```
Qu'est-ce que vous remarquez dans les fichiers `passwd` et `shadow`? Lequel de ces deux fichiers change? Pourquoi ? Où se trouve donc l'information du mot de passe? Quelles sont les permissions du fichier `shadow` et pourquoi ?

**d)** Changez à nouveau le mot de passe du même utilisateur et donnez-lui le ***même*** mot de passe. Est-ce que les informations du mot de passe ont changé? Pourquoi?

**e)** Créez un deuxième utilisateur en suivant les mêmes étapes qu'au point b. Éditez ensuite le fichier `shadow` et remplacez la valeur par défaut (!!) du champ de mot de passe de l'utilisateur que vous venez de créer par la valeur du même champ pour l'utilisateur que vous avez créé en premier (les éditeurs de texte nano et vim sont disponibles). Sauvegardez le fichier et quittez votre session. Essayez de vous connecter sur le compte du deuxième utilisateur mais avec le mot de passe que vous venez de copier. Est-ce que ceci est possible? Expliquez pourquoi. Quel est le problème?

**f)** Effacez cet utilisateur avec la commande
```
$ userdel -r NOM
```
Qu'est-ce qui se passe dans `passwd` et `shadow` ?

## Question 5 - Contrôle de qualité de choix de mot de passe [/1]

Dans cette question, vous allez vous familiariser avec l'outil « John The Ripper » qui permet de trouver des mots de passe.

Le principe de base est très simple : en regardant les entrées dans un fichier `/etc/passwd`, « John » essaie des mots dans un dictionnaire de mots de passe courants et génère des hash avec les « salt » retrouvés et les compare avec les hash de mots de passe dans ce fichier. Il utilise des fichiers `/etc/passwd` à l'ancienne (qui contiennent aussi le hash de mot de passe). Cependant, il est possible de reconstruire ce type de fichier à partir du `/etc/passwd` et du `/etc/shadow` en utilisant la commande « unshadow » qui vient avec John The Ripper. Pour cette question, ceci a déjà été fait pour vous et vous disposez donc de 2 fichiers «password1» et «password2» provenant de deux machines d'un même sous-réseau. Ils sont dans les répertoires **/root/**.

**a)** Ouvrez une session sur la machine virtuelle **Mdp** en utilisant le compte `root` et le mot de passe **`Af6Thg+9`**. John est utilisé en lançant la commande « `john` » à partir du répertoire `/etc/john/` avec un fichier du type `/etc/passwd` ancien en argument :

```
$ cd /etc/john
$ john Nom_Fichier
```

(*avec Nom_Fichier le chemin complet du fichier*)

Familiarisez-vous avec l'outil et essayez de trouver le plus de mots de passe possible sur les fichiers password1 et password2. Vous pouvez à tout moment voir ce que John est en train de faire en appuyant sur la barre d'espacement. Les résultats de John sont répertoriés dans le fichier **john.pot**. En examinant ce fichier (ou en regardant les résultats sur la console), décrivez quels mots de passe vous avez trouvés, à quels usagers ils correspondent et sur laquelle des deux machines (1 et 2) ils se trouvent. Joignez une trace du fichier john.pot ou des résultats de la console à votre rapport.

Ne vous attardez pas trop, car certains mots de passe peuvent prendre beaucoup de temps et demanderaient l'utilisation de John au niveau 3. Un quart d'heure est largement suffisant pour détecter les mots de passe des niveaux 1 et 2.

**b)** Calculez l'entropie maximale pour les alphabets suivants :

- [a-zA-Z]
- [a-zA-Z0-9]
- L'ensemble de la table ascii

**c)** Déduisez des résultats de b) un critère important pour qu'un mot de passe soit fort.

**d)** Au vu des résultats de John the Ripper, donner 3 autres critères pour qu'un mot de passe soit fort.

# Partie C

Pour les deux questions suivantes, il ne faut répondre qu'une fois par équipe. Choisissez la matricule d'un des membres pour obtenir le texte chiffré.

## Question 1 - Échec du protocole RSA [/0.75]

Malgré le fait que RSA soit considéré sécuritaire, si n est assez grand pour ne pas être factorisable, il faut tout de même faire attention.

Bob utilise le chiffrement RSA avec un n assez grand pour ne pas être factorisable. Alice envoie à Bob un message dans lequel chaque caractère alphabétique est chiffré séparément avec la clé publique de Bob, les caractères avant le chiffrement étant représentés par des nombres de 0 à 25 ('A'=0, 'B'=1,..., 'Z'=25).

a) Décrire comment Ève peut facilement déchiffrer ce message.

b) Récupérer le texte à déchiffrer et la clé publique dans le document INF4420A_TP1_Q1_H15 du site Moodle. Utilisez l'attaque décrite précédemment pour déchiffrer le texte, sans factoriser n, selon la clé. Donnez votre réponse en texte, pas en chiffres.

c) Si vous regardez attentivement la liste de textes à déchiffrer pour votre groupe de laboratoire, vous remarquez probablement des textes chiffrés avec des « 0 » ou des « 1 ». Quelles conclusions additionnelles pouvez-vous tirer sur le contenu des messages pour assurer le bon fonctionnement de RSA ?

## Question 2 - Déchiffrement "simple" [/0.75]

Vous êtes en possession d'un extrait de 200 caractères chiffrés qui provient d'un texte en anglais qui vous est inconnu. Rien n'indique que le début de cet extrait est un début de mot et que la fin est une fin de mot. Il y a 27 caractères possibles dans l'extrait chiffré que vous possédez, soient les 26 lettres de l'alphabet et le caractère « @ ». Ceci correspond à un alphabet en clair composé des 26 lettres de l'alphabet et de l'espace. Il est également important de mentionner qu'une fin de phrase est représentée par deux espaces dans le texte original, soit avant que le chiffrement ait été effectué.

Récupérer le texte chiffré qui vous a été assigné et que vous devez déchiffrer dans le document INF4420A_TP1_Q2_H15 du site Moodle. Deux outils sont mis à votre disposition pour effectuer le déchiffrement :

- Applet aidant au déchiffrement d'une substitution simple : http://www.cs.trincoll.edu/~crypto/cryptogrammer/
- Utilitaire `frequency` (dossier « Source – Entropie - Chiffrement ») pour calculer le nombre d'occurrences des caractères dans un texte. L'option –n permet de spécifier la taille de bloc. Dans le cas où celle-ci est omise, une taille de 1 sera utilisée.

Vous devez fournir le texte déchiffré.

Finalement, pour faciliter votre tâche de déchiffrement, les fréquences des lettres en anglais vous sont fournies dans le tableau suivant :

| Lettre | Frequence | Lettre | Frequence |
|---|---|---|---|
| e | 12.702% | m | 2.406% |
| t | 9.056% | w | 2.360% |
| a | 8.167% | f | 2.228% |
| o | 7.507% | g | 2.015% |
| i | 6.966% | y | 1.974% |
| n | 6.749% | p | 1.929% |
| s | 6.327% | b | 1.492% |
| h | 6.094% | v | 0.978% |
| r | 5.987% | k | 0.772% |
| d | 4.253% | j | 0.153% |
| l | 4.025% | x | 0.150% |
| c | 2.782% | q | 0.095% |
| u | 2.758% | z | 0.074% |

**Figure 5.1 : Fréquences relatives des lettres en anglais**[1]

Il est également à noter que l'espace est 1.07 fois plus fréquent que la lettre e en anglais[2].

Les digrammes les plus courants en anglais sont, en ordre[3] :

th, he, in, en, nt, re, er, an, ti, es, on, at, se, nd, or, ar, al, te, co, de, to, ra, et, ed, it, sa, em, ro.

Les trigrammes les plus courants en anglais sont, en ordre[4] :

the, and, tha, ent, ing, ion, tio, for, nde, has, nce, edt, tis, oft, sth, men.

1 Source : http://en.wikipedia.org/wiki/Letter_frequencies

2 Source : http://en.wikipedia.org/wiki/Letter_frequencies

3 Source : http://pages.central.edu/emp/LintonT/classes/spring01/cryptography/letterfreq.html

4 Source : http://pages.central.edu/emp/LintonT/classes/spring01/cryptography/letterfreq.html