# Partie B

## Question 1 – Codage

1. **Expliciter les alphabets , et qui sont respectivement les alphabets pour la sortie de la source, du codeur et du bloc de chiffrement**

L’alphabet de la sortie de la source est les chiffres de [0-9].

L’alphabet de la sortie du codeur est le bit, donc il prend la valeur [0-1].

L’alphabet de sortie du bloc de chiffrement donne aussi le bit comme alphabet puisqu’il ne fait que faire des opérations sur les bits.

1. **Identifiez** **les langages provenant des alphabets , et** .

Le langage provenant de la source est une suite de quatre chiffres.

Le langage provenant de la sortie du codeur est deux fois la même séquence de 32 bits.

Le langage provenant de la sortie du bloc de chiffrement donne une plage de 64 bits.

1. **Identifiez les attaques auxquelles le système est vulnérable.**

Un tel système est vulnérable aux attaques de forces brutes. Malgré la longueur de la sortie du bloc de 64 bits, il y en en fait que 10000 combinaisons, soit les quatre chiffres. Une seconde attaque est d’enregistrer un message intercepté et de le réexécuter plusieurs fois.

1. **Montrez à l’aide de traces d’exécution comment vous les effectueriez.**

Le script nommé « BruteForce.py », dans le dossier scripts, permet de générer un dictionnaire avec les 10000 combinaisons possibles. Ainsi, lorsqu’on intercept un message il suffit de faire une recherche dans le dictionnaire et de compter le numéro de ligne associé.

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

|  |  |
| --- | --- |
| **Codage** | **Avantage** |
| Codage 1 | Rend beaucoup plus difficile l’attaque de force brute parce qu’il y a une longue chaine de caractère aléatoire à la fin des messages. |
| Codage 2 | Rend plus difficile l’attaque de force brute et empêche la réexécution de la requête grâce au « Timestamp ». |
| Codage 3 | Empêche la réexécution de la requête. |

1. **Selon vous quel est le meilleur codage ? Pourquoi ?**

Le codage 2 est le meilleur codage car il rend difficile l’attaque brute et empêche la réexécution de la requête. Il s’agit du seul des trois codages qui agit sur les deux attaques simultanément.

## Question 2 – Exploitation d’une vulnérabilité critique

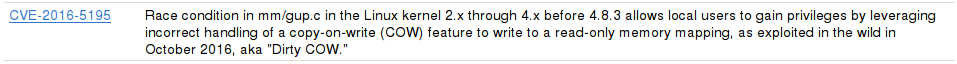
1. **Quelle est la version du noyau utilisée par la machine virtuelle? Donnez la commande utilisée ainsi que sa sortie.**

La version du noyau utilisée par la machine virtuelle est 3.4.5-harden. Cette information a été récupérée par la ligne de commande : « *uname* *-r* ». *Uname* (unix name) permet d’afficher les informations système de la machine et l’argument *r* représente le *release*.

C:\Users\Etienne\AppData\Local\Microsoft\Windows\INetCache\Content.Word\q2-a.png

1. 1. **Quel est l’identifiant de la faille « Dirty Cow » (commençant par CVE-2016) ?**

L’identifiant de la faille est le 5195.



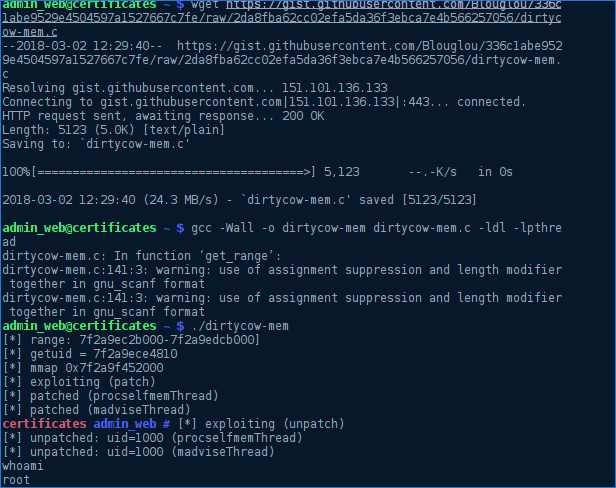
2. **Expliquez en quelques lignes cette faille de sécurité. Pourquoi est-elle aussi critique?**

Il s’agit d’une faille dans le système de mémoire du noyau qui gérait la copie en écriture (copy-on-write COW). Cette faille permet à un attaquant disposant d'un compte système local de modifier les binaires sur le disque, en ignorant les mécanismes d'autorisation standard qui empêcheraient toute modification sans un ensemble d'autorisations approprié.

3. **Notre système est-il vulnérable à cette faille? Pourquoi?**

Notre système est vulnérable car sa version de noyau est de 3.x et cette faille de sécurité affecte les noyaux de 2.x à 4.x.

1. **L’exploit de « Dirty Cow »**

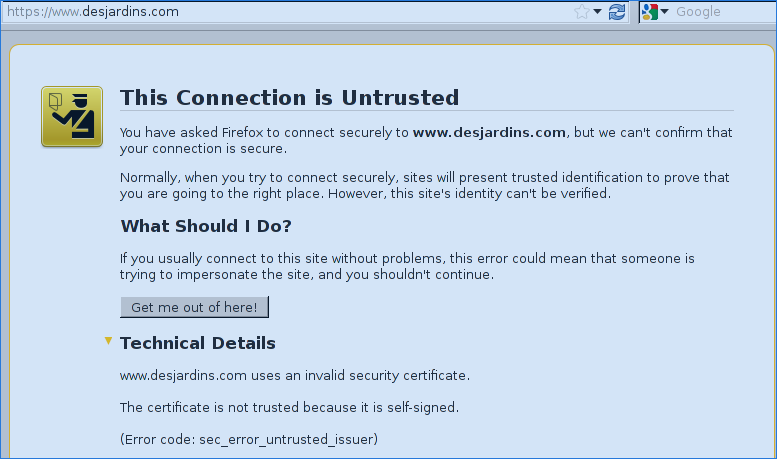
**C:\Users\Etienne\AppData\Local\Microsoft\Windows\INetCache\Content.Word\q2-c-6.png**

## Question 3 – Certificats à clé publique, HTTPS et SSL

1. **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?**

La page est bloquée car on ne peut pas confirmer que la connexion est sécuritaire en raison que le certificat n’est pas signé par une autorité reconnue.

1. **Qu'est-ce** **qui pourrait vous aider à découvrir que le site est une fraude?**

Le navigateur web bloque la connexion et affiche un message d’avertissement.

1. **Quel** **est maintenant le nouveau comportement de Firefox et pourquoi?**

Le navigateur web permet d’accéder à la page souhaitée parce que nous avons mentionné que nous avions confiance à l’autorité qui signe les certificats.

1. **Quel est le comportement de Firefox et pourquoi?**

Le navigateur web permet l’accès au site web car <https://wwww.desjardins.com> est maintenant signé avec un certificats CA dont nous sommes l’autorité.

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

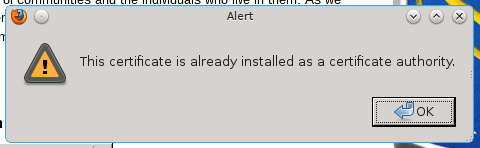
Les deux sites web sont bloqués car leur certificat n’est pas signé par une autorité reconnue.

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

Nous avons ajouté le certificat de façon temporaire. Ainsi, l’autorité de certification est acceptée que pour une session de navigation. Ainsi, lorsqu’on efface le cache de Firefox les permissions sont enlevées et le site est redevenu bloqué.

1. **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.**

Lorsqu’on tente d’ajouter le certificat de l’autorité du site de [www.rbc.com](http://www.rbc.com) on nous affiche se message nous mentionnant que nous avons déjà autorisé cette autorité de certification.



Le site [www.rbc.com](http://www.rbc.com) n’est plus bloqué car le navigateur l’ajouté comme source fiable de façon permanente car nous avons ajouté deux fois cette autorité.

1. **Aller sur www.bmo.com. Que se passe-t-il ? Expliquez.**

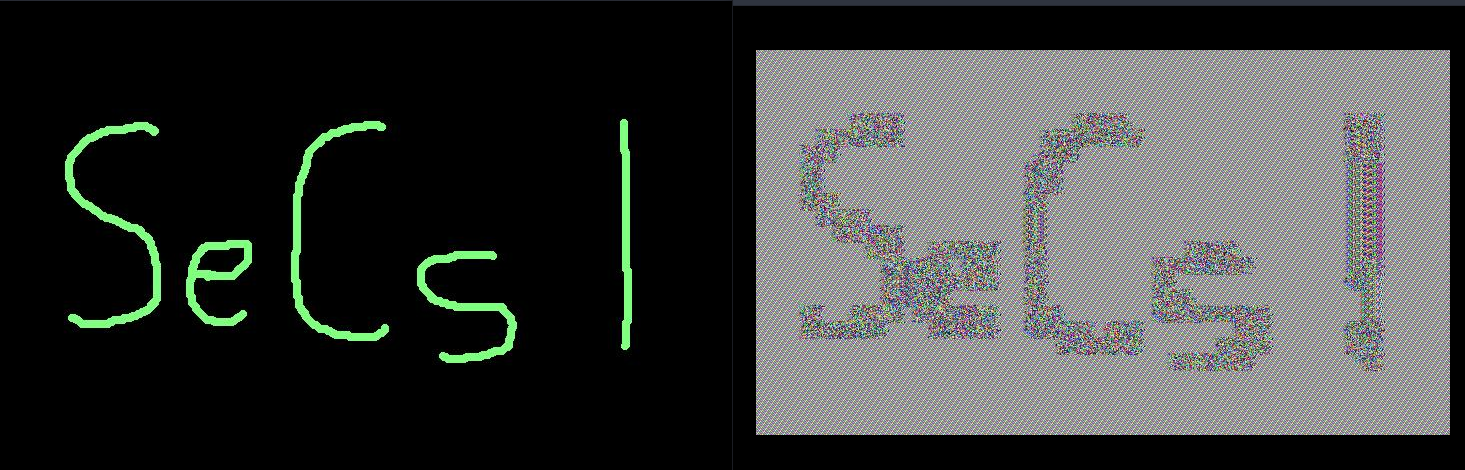
Le site de bmo n’est pas bloqué puisque l’autorité de certification « Verisign Inc. » signe aussi le site de rbc. Donc, dès que l’on approuve l’autorité pour un site, tous les sites qui ont le certificat de cette autorité sont approuvés.

1. **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?**

Il est dangereux d’accepter les certificats « self-signed », car il n’y a aucune autorité qui a autorisé qui a approuvé ce site. Pour ce qui est des certificats CA, il est possible que l’autorité signe des sites légitimes, cependant il se peut que certains sites web soient malicieux. Donc, lorsqu’on fait confiance à une autorité on fait confiance à tous les sites que celle-ci signe.

## Question 4 – Chiffrement par bloc et modes d’opération

1. **À l’aide du script python AES.py, chiffrez ce fichier en mode ECB. Observez le fichier de sortie et commentez.**

****

À gauche le fichier original et à droite le fichier chiffre en mode ECB. Le fichier chiffré, bien qu’il soit modifié, est encore très lisible et très peu sécuritaire. Le fonctionnement de « Electronic codebook » est que le message à chiffrer est subdivisé en plusieurs blocs qui sont chiffrés séparément les uns après les autres. Le gros défaut de cette méthode est que deux blocs avec le même contenu seront chiffrés de la même manière, on peut donc tirer des informations à partir du texte chiffré en cherchant les séquences identiques.

1. **Chiffrez maintenant le fichier en mode CBC. Observez le fichier puis commentez.**



Le chiffrement par CBC est totalement illisible et il est impossible de déchiffrer le message à l’œil nue. Le « Cipher Block Chaining » applique sur chaque bloc un OU exclusif avec le chiffrement du bloc précédent avant qu’il soit lui-même chiffré. De plus, afin de rendre chaque message unique, un vecteur d'initialisation est utilisé. Donc, chaque bloc a un impact sur le bloc précédent.

1. **Concluez sur l’importance des modes d’opération des algorithmes de chiffrement par bloc.**

Le choix d’un bon mode de chiffrement pour les algorithmes de chiffrement par bloque est crucial. Pour qu’on algorithme soit efficace il faut que chaque bloc à chiffrer impacte le suivant. Ainsi, il est difficile de voir des tendances dans le message chiffré.

# Partie C

## Question 1 - Échec du protocole RSA

1. **Décrire comment Ève peut facilement déchiffrer ce message.**

Le codage de ce message est faible. Il y a 26 caractères et ils sont codés à une correspondance de 1 pour 1. Ainsi, chaque lettre correspond à un seul nombre dans la possibilité de n. Donc, même si le n est assez grand, il est facile de trouver les 26 nombres et ainsi faire des correspondances.

1. **Donnez votre réponse en texte, pas en chiffres.**

Avec le fichier rsa.py, nous avons réussi à déchiffrer le message.

Matricule : 1773922

e = 311

n = 288419

|  |  |
| --- | --- |
| Lettre chiffré | Lettre claire |
| 0 | A |
| 81902 | R |
| 81902 | R |
| 71381 | E |
| 139280 | T |

Le message est donc ARRET.

1. **Quelles conclusions additionnelles pouvez-vous tirer sur le contenu des messages pour assurer le bon fonctionnement de RSA ?**

Les textes chiffrés contenant des 0 ou des 1 signifie qu’ils possèdent les lettres A ou B puisque :

Il faut donc que le codeur empêche d’utiliser les chiffres 0 et 1 car ils restent intacts.

## Question 2 - Déchiffrement "simple"

Le script decode.py permet de décoder le code du matricule 1773922.

Le message chiffré : TUQQUZRBZCYGZZZSQZBERQUZTLBZBLZGQQOZBLZGQQZBSQZPLL@GZLEZSQYEZBSQZEQOYE@GZYGZSQZKYGGQUZYPLTNZVHBZBSQJZCQEQZILLUZYTUZUERT@ZBLZSROZZZGOYPPQEZVLJGZBSYTZSROGQPIZIPLM@QUZYBZSRGZSQQPGZYGZKELHUZBLZVQZGQQTZCRB

Le message déchiffré :

NDEED IT WAS. HE TRIED NOT TO SEEM TO SEE THE LOOKS OR HEAR THE REMARKS AS HE PASSED ALONG BUT THEY WERE FOOD AND DRINK TO HIM. SMALLER BOYS THAN HIMSELF FLOCKED AT HIS HEELS AS PROUD TO BE SEEN WIT

|  |  |
| --- | --- |
| **Lettre chiffré** | **Lettre correspondante** |
| @ | K |
| B | T |
| C | W |
| E | R |
| G | S |
| H | U |
| I | F |
| J | Y |
| K | P |
| L | O |
| M | C |
| N | G |
| O | M |
| P | L |
| Q | E |
| R | I |
| S | H |
| T | N |
| U | D |
| V | B |
| Y | A |
| Z | (Espace) |