UQAR

Université du Québec à Rimouski

300, allée des Ursulines, C. P. 3300, succ. A
Rimouski (Québec) G5L 3A1, CANADA
www.uqar.ca

DÉPARTEMENT DE MATHÉMATIQUES, D'INFORMATIQUE ET DE GÉNIE

Sécurité Informatique
Devoir 1 — Énoncé

SIGLE :	INF36207
TITRE :	Sécurité Informatique
GROUPE :	06
PROFESSEUR :	Steven Pigeon
	K-212
	steven_pigeon@uqar.ca
DATE DE REMISE :	12 Février 2019, avant minuit

---

0. Modalités. Vous devez faire le devoir en équipes de deux. Le devoir devra être remis par courriel, lequel contiendra une archive comprenant un document texte (.docx ou autre) qui contient les réponses (discussions, numéro bonus et mots de passes récupérés) et les programmes que vous devrez réaliser au n° **2**. Les programmes devront être fonctionnels, évidemment, et l'archive devra contenir tout ce qui est nécessaire pour les faire fonctionner (on doit pouvoir les compiler et les exécuter). Le nom de l'archive doit contenir le nom des deux coéquipiers et le numéro du formulaire : 2 . Les versions électroniques (pdf) seront aussi sur Moodle.

Notez que chaque devoir est *unique*. Le numéro 2 ne peut être utilisé que par une seule équipe. Si deux équipes utilisent le même formulaire, les deux seront disqualifiées — un euphémisme pour « auront zéro ».

1. Casser des mots de passe. 15 pts. Supposez que vous ayez capturé le fichier de mots de passe d'un système dont vous tentez de prendre le contrôle. Vous voulez obtenir les mots de passe originaux : vous devez monter une attaque contre ces mots de passe qui sont camouflés grâce à l'algorithme MD5. Les *hashes* que vous avez capturés sont les suivants :

```
bd787548b57d899d882ae21ca8e75263
1ed001839e6709c59e1ca1357952d8f5
27eacbc30f7fde770cc274bdb20bc186
e0be4fd60faa2f3d904dc237eb2647a9
3961a903097f9cb79f07dae17bac2f11
4130a0993d4c2e9486768e887104b980
6268dda92c28ed9a39024fc6cfd360e9
38ffed95337f0cd8a1870ac111533671
7994f0b70ac711650b7c6a6e2ab47c65
beffd790d7dedae2a68d507932b2732c
7635a53a1e8308e8452a33cc1595316a
c69659150abcc53287dd263361b86077
b9a6425b8b5b41a8a0d56d1e6fbf4b95
72aa6cfbe3f8a385e8391303eac0f34f
2716c1e5bee3a6d31e720aab9d769a8d
```

Pour ce qui suit, vous avez le choix du langage de programmation : Java, C, C++, C#, Python, Windows-Power Shell, Bash, etc. Cependant, pour le numéro **1 b)**, considérez un langage capable de générer du code *rapide*. De plus, les librairies qui calculent le hash MD5 sont disponibles sur toutes les plateformes et il en existe des *bindings* pour la plupart des langages, vous n'aurez donc pas à l'implémenter vous même. Quelque soit le langage que vous aurez choisi, vous devez remettre un projet complet qui montre que le code fonctionne — *pas de code, pas de points*.

a) Attaques par dictionnaire. 3 pts. Utilisez le dictionnaire `mots-8-et-moins.txt` (que vous trouverez sur Moodle) pour monter une attaque par dictionnaire contre les *hashes* ci-dessus. Donnez les mots de passe en clair récupérés.

b) Attaques par énumération. 10 pts. Vous avez remarqué que certains des *hashes* ne sont pas résolus par l'attaque par dictionnaire. Dans ce cas, vous devrez monter une attaque par énumération où vous allez générer toutes les chaînes de longueur 1, toutes les chaînes de longueur 2, toutes les chaînes de longueur 3, etc. jusqu'à ce que vous trouviez une chaîne dont le *hash* corresponde. Puisque le but n'est pas de vous faire passer de longues heures de temps de calcul, supposez que les mots de passe n'ont pas plus de 8 caractères de long, et que les caractères sont tirés de l'alphabet suivant :

```
abcdefghijklmnopqrstuvwxyz0123456789!@#$%&*
```

Donnez les mots de passe récupérés en clair.

2. Sur la complexité de casser les mots de passe. 2 pts. À la lumière du n° **2**, qu'est-ce que vous pouvez dire sur le choix d'un mot de passe ? Qu'est-ce que vous suggéreriez pour déjouer à la fois les attaques par dictionnaire et les attaques par énumération ? Quelle stratégie *simple* utiliseriez vous pour vous choisir un mot de passe sécuritaire (sans pour autant être impossible à taper) ?