1 Overall marking scheme

The coursework for COMP522 consists of two assignments, contributing to 25% of the final mark. The contribution of the single assignments is as follows:

<table>
<thead>
<tr>
<th>Assignment</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assignment 1</td>
<td>12.5%</td>
</tr>
<tr>
<td>Assignment 2</td>
<td>12.5%</td>
</tr>
<tr>
<td>TOTAL</td>
<td>25%</td>
</tr>
</tbody>
</table>

Failure in any assignment may be compensated for by higher marks in other components of the module.

This document describes Assignment 1. Assignment 1 will be marked according to the following broad criteria:

- correctness of the program;
- presence/absence of the report on the experiments;
- clarity of the arguments explaining the observed behaviour.

2 Aims of the Assignment 1

- to illustrate the practical complexity of brute-force search attacks on the password-based encryption;
- to test the students skills of using symmetric cryptography primitives in Java programmes;
- to test the students skills in the analysis of the experiments.
3 Brute-force search attack on the password-based encryption

This exercise asks you to write a program implementing password-based encryption and decryption, and then to extend it with the class(es) implementing brute-force search attack. You need to

- implement a program which takes an user password as the input and performs encryption of the predefined plaintext; then it asks the password again and decrypt the ciphertext;
- extend your program with the class(es) implementing brute-force search attack on your encryption/decryption procedure;
- the attacker knows:
  - the predefined plaintext;
  - the ciphertext produced;
  - the salt;
  - the iteration count;
  - but no password.
- thus an attacker should iterate over all passwords up to the given length \( n \), encrypt the plaintext and compare the result with the given ciphertext;
- assume for simplicity, that the password consists of the decimal digits 0, . . . , 9 only;
- estimate average time required to find a correct password for the predefined plaintext/ciphertext, fixed value of the salt and small values of \( n \);
- investigate how the search time depends on the iteration count value.
- formulate your recommendation on the length of the password, which would allow to withstand brute-force search attack for one hour (1 hour);
- consider a variant of the attack, in which an attacker knows everything as above, except the iteration count, and estimate the time required to recover the password of the small length \( (n=3, 4) \).

4 Useful information

You may find it useful to have a look on the simple program implementing password-based encryption:

http://www.csc.liv.ac.uk/~alexei/COMP522/PBEs.java

5 Submission

You need to submit:

- Java code and compiled classes of your program
- short report on experiments

The work must be submitted electronically by going to the Web page at https://cgi.csc.liv.ac.uk/login.php and follow the link “Coursework submission.” This must be done by

17.00 on Monday November 6, 2017

Please be aware that the standard University policies

- on plagiarism, collusion and fabricated data
  www.liv.ac.uk/tqsd/pol_strat_cop/cop_assess/cop_assess.doc, Section 8 and

- on late submission
  www.liv.ac.uk/tqsd/pol_strat_cop/cop_assess/cop_assess.doc, Section 6

are applied to this assignment.