Home Assignment 1

Advanced Web Security

2016

B-assignments

Complete the four B-assignments below and solve them in groups of two students.

B-1 Implement the Luhn algorithm in your favorite language. You will be given a list of card numbers with one digit censored with an "X". Use your implementation to find the censored digit in order for the card number to be valid. The list consists of 100 card numbers, one per line. The answer is the concatenation of all censored digits, in order according to the list.
Example:

```
12774212857X4109
586604X108627571
7473X86953606632
4026467X45830632
20X3092648604969
```

The answer to the example above is "54963".
Assessment:

- Upload your code to Urkund, paul.stankovski.lu@analys.urkund.se.
- There will be one Moodle question following the problem statement above. There will be a test quiz on Moodle, where you can try your implementation as many times as you like. The test quiz will not be graded.
- There will be one Moodle question regarding the Luhn algorithm in general.

B-2 Read sections 4 and 4.1 in the MicroMint paper[1]. Write a program that simulates the time needed to generate MicroMint coins. Model the process as a balls-and-bins problem. Your program should take three parameters; u, k and c. The parameter u is the number of bits used for identifying the bin, so there are $b = 2^u$ bins. Parameter k is the number of collisions (balls in the same bin) needed to make a coin, so that fewer than k balls in a bin make no coins, and k or more balls in a bin make precisely one coin. All bins are empty at first. At each iteration, throw a ball into a randomly selected bin. How many iterations (how many balls thrown) does it take before you generate $c = 1, 100$ and 10,000 coins, respectively? You will be given u, k and c.
Note: With respect to the MicroMint paper, use $t = 0$.
The above procedure describes one simulation. You will need to run several such simulations in order to compute **the averages** reliably. Construct 99.9% confidence intervals for the three mean values. So, you may ask, for how long do I need to run these simulations? You will be given appropriate widths for your confidence intervals. With n statistical observations $x_1, x_2, \ldots, x_n$, a 99.9% confidence interval for the mean is given by

$$\left(\bar{x} \pm \lambda \cdot \frac{s}{\sqrt{n}}\right),$$

[1] R. L. Rivest and A. Shamir - PayWord and MicroMint: Two simple micropayment schemes

where $\bar{x}$ and s are your sample-estimated average and standard deviation, respectively. You may use $\lambda = 3.66$. Below we present the minimum and maximum values for u and k that you will be able to handle. **Note**: A simulation should take no longer than 30 minutes.
Example: In the tables below you are given two examples of simulations results. In the first example, the simulation is run with the parameters $u = 16, k = 2, c = 1$. The mean is calculated to be 322. Note that this value is just an estimate of the *true* mean value. If you create a confidence interval of width 22 (in this case) for the mean, then you will reliably assess the true mean value, which will be enough to pass this assignment.

Table 1: Simulation with $u = 16, k = 2$.

# coins (c)	mean ($\bar{x}$)	c.i. width
1	322	22
100	3685	24
10k	45270	22

Table 2: Simulation with $u = 20, k = 7$.

# coins (c)	mean ($\bar{x}$)	c.i. width
1	493981	79671
100	1069997	15616
10k	2420113	4783

Assessment:

- Upload your code to Urkund, paul.stankovski.lu@analys.urkund.se.
- There will be one Moodle question following the problem statement above. There will be a test quiz on Moodle, where you can try your implementation as many times as you like. The test quiz will not be graded.
- There will be one Moodle question regarding MicroMint in general (Sections 4 and 4.1 in the paper).

B-3 Compare the security between SET and 3D Secure in terms of authentication, encryption, cardholder, verfication etc.
Assessment:

- Write a short (< 0.5 A4 page) report on the subject.
- Upload your report to Moodle (it will be manually graded).

B-4 When the number of transactions in a Bitcoin block is very large, the Merkle tree can be used to prove that a specific transaction is in that block without the need of revealing the entire transaction set, thereby minimizing data transfer. Illustrate this and explain how the Merkle tree of a transaction set can be used to this end. How many hashes must be downloaded to prove that a specific transaction is in a given block with 2^{10} transactions?
Assessment:

- Write a short (< 0.5 A4 page) report on the subject.
- Upload your report to Moodle (it will be manually graded).

C-Assignments

Complete one out of the two C-assignments below and solve them in groups of two students.

C-1 Implement the coin withdrawal (the version with 2k quadruples) in the untraceable e-cash scheme given in the lecture notes.

- You may use a variant of RSA with easily manageable numbers.
- The data transfer can be simulated locally.
- The extended Euclidean algorithm can be used to find the inverse of 3 mod n.
- The square-and-multiply technique can be used to efficiently compute the signature.
- Choose sensible functions f and h.

Assessment:

- Summarize your work in a short report, making it clear that the program works as intended.
- Upload your report to Urkund, paul.stankovski.lu@analys.urkund.se.
- Upload your report to Moodle (it will be manually graded).

C-2 The Payment Card Industry Data Security Standard (PCI DSS) has been developed to increase the security of card transactions. Read about PCI DSS and summarize the standard in approximately 3 pages. Make sure you cover its purpose and goals, who it applies to, and a summary of the requirements.
Assessment:

- Upload your report to Urkund, paul.stankovski.lu@analys.urkund.se.
- Upload your report to Moodle (it will be manually graded).