Luhn Algorithm

John McCall

December 9, 2013

1   Backround

The Luhn algorithm, which is also called the modulus 10 algorithm, implements a simple checksum formula and is used in validating identification numbers. Credit card numbers, IMEI numbers, National Provider Identifier numbers, and Canadian Social Insurance Numbers are a few examples of the types of ids that the Luhn algorithm is used with. It was created by Hans Peter Luhn, a scientist working for IBM, in 1954.

This algorithm is meant to detect accidental errors, as opposed to malicious attacks, so by design it is not a cryptographically secure function. It is mostly used as an easy way to distinguish valid numbers from a series of random digits.

2   The Algorithm

The Luhn Algorithm verifies a numbers using its check digit, which is usually added to a partial account number to generate the whole number. There is a test that the account number must pass:

1. From the rightmost digit, which is the chech digit, moving left, double the value of every second digit; if the product of this doubling operation is greater that 9(e.g., $7*2 = 14$), then sum the digits of the products (e.g., 10: $1 + 0 = 1$, 14: $1 + 4 = 5$).

2. Take the sum of all the digits.

3. If the total modulo 10 is equal to 0 (if the total ends in zero) then the number is valid according to the Luhn formula; else it is not valid.

3   Example

Take the account number 7992739871, we will add a check digit, making the number: 7992739871x.

We obtain x by computing the sum of the digits and then multiplying that value by 9 modulo 10. In this case, the sum of the digits is 67. 67 times 9 equals 603, modulo 10 gives us 3, which is the check digit.

Using Luhn's Algorithm on the whole account number (79927398713) should result in a 0, meaning that the account number is valid. Alternately, if you already know the check digit, one can do the computation to determine the check digit and compare that result to the known digit. If they match the account number is valid, otherwise it's invalid.

4   Strengths and Weaknesses

The Luhn algorithm will catch any single-digit error and most cases of switching adjacent digits. However, it will not detect 09 being switched to 90 (or vise versa). It will detect 7 of the 10 possible twin errors (it will not detect: $22 \leftrightarrow 55, 33 \leftrightarrow 66, 44 \leftrightarrow 77$).