**Pre-image and Collision Attacks on the SHA-1 Hashing Algorithm**

Stephen Howard

A cryptographic hash function is an algorithm that maps data of arbitrary size (referred to as the message) to an integer representation (often called a hash value or a digest) of a predefined size. Ideally, this process is a one-way function, such that someone that has only the hash value cannot use this value to determine the data that produced it. Additionally, it should be computationally impossible for a user to find two pieces of data that produce the exact same hash value.

SHA-1 is one of these hash functions, and it produces a 160-bit hash value or digest from a message. For many years, this algorithm was the standard for cryptographers. However, since 2005, SHA-1 has not been considered secure since some vulnerabilities have been found with it. These vulnerabilities include the relative ease of performing pre-image attacks and collision attacks against the algorithm. As an example of these vulnerabilities: Google announced they had performed a successful collision attack against SHA-1 in February 2017.

Pre-image attacks refer to attacks where the attacker is given an existing hash output of some data. The goal of the attacker is then to find a different piece of data that hashes to that same digest. The process for this attack is usually to generate a digest on some new data, compare it to the goal, and repeat until a new digest is found that matches the goal. This process takes approximately 2n guesses, where n is the size of the digest, in bits (160 for SHA-1). For a full SHA-1 hash system, it would take on average 1.46 x 1048 attempts for this attack to be successful.

Collision attacks are attacks where the goal is to find any two messages that map to the same hash value. This is done by generating a hash of some data, and then storing that hash value. Another hash is then generated, at which point it is compared with all stored hash values. If it doesn’t match any digests in the store, it is stored, and the process is repeated. Once any match is found, the attack is successful. This is significantly easier computationally than a pre-image attack, since in this instance the goal is to find *any* match, not a specific one. On average, this attack requires 2n/2 attempts. So, for the 160-bit SHA-1 system, it would take around 1.21 x 1024 guesses.

For both of these attacks, the difficulty of a successful attack increases exponentially with the size of the digest generated. To test this idea, I ran 50 experiments for 8-, 10-, 12-, 14-, 16-, 18-, 20-, and 22-bit digests for both pre-image attacks and collision attacks. These smaller digests were obtained by finding a SHA-1 hash for the string and then truncating the bits to the requested length. My experimental results, along with the theoretical results, are shown in the table below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Average # of Attempts Until Successful Attack | | | | |
| Digest Length (n) | Pre-image | | Collision | |
| Experimental | Theoretical (2n) | Experimental | Theoretical (2n/2) |
| 8 | 293.74 | 256.00 | 20.10 | 16.00 |
| 10 | 685.96 | 1,024.00 | 39.78 | 32.00 |
| 12 | 4,553.50 | 4,096.00 | 81.36 | 64.00 |
| 14 | 18,402.38 | 16,384.00 | 153.08 | 128.00 |
| 16 | 45,061.12 | 65,536.00 | 330.76 | 256.00 |
| 18 | 331,986.88 | 262,144.00 | 708.36 | 512.00 |
| 20 | 1,064,303.42 | 1,048,576.00 | 1,234.12 | 1,024.00 |
| 22 | 3,718,136.20 | 4,194,304.00 | 2,432.34 | 2,048.00 |

Due to the nature of this experiment, it is very possible that any one trial will take substantially more, or fewer, attempts than is predicted. However, this is balanced out quite nicely by running 50 experiments of each attack type/digest length combination. As we can see in the table, the experimental results are remarkably close to what we had expected (the ‘theoretical’ columns). For the pre-image attacks, the experimental data is higher than anticipated at some digest sizes, and lower at others. For the collision attacks, the experimental data is consistently slightly higher along all digest sizes, however the variance is not enough to raise concerns. We can confidently say that the theoretical speed estimates (2n for pre-image, 2n/2 for collision) are accurate representations of reality. Below, these results are graphed on a logarithmic scale.

In conclusion, one of the most important aspects of a hashing algorithm is the size of the output digest. Simply put, the larger the digest size, the more possible outputs there are, so the likelihood of two inputs having the same output is diminished. My results strongly confirm this concept. This explains why most cryptographical hash functions introduced after SHA-1 produce larger digests.

**Reviewed by Ed Ringger, BYU ACME graduate (CS minor)**