Matt Christensen (mrc621)

September 17, 2021

CS 465 (001) – Clift, Frederic M

Project #2: Hash Attack

Introduction

An important security principle is hashing. The idea of hashing is to take an input of any length and create a fixed length digest as output – that represents the input. Hashing algorithms attempt to create unique output for each input wherever possible. This relationship between the input and the hash needs to have a few characteristics to be useful within security. For example, because there are infinite inputs (because inputs can be of arbitrary length) and finite outputs there will be times that two given inputs will have the same output (called hash collisions). It is important to try and minimize these collisions. Another property of a good hashing algorithm (that is relevant for this paper) is the ability to be one-way: meaning it is computationally infeasible to find an input that maps to a given hash/digest.

The purpose of this report is to compare and contrast *collision* and *pre-image hash attacks* through experimentation, which target the stated properties of a good hashing algorithm. Specifically, simple experiments (in the form of basic hash attacks utilizing SHA-1) will be used to compare theoretical and practical costs for hashes of various bit lengths. These experiments will demonstrate the distinction between *collision* and *pre-image attacks* and their hypothetical probabilities of success.

Experiment

The experiment consists of two different hash attacks – the first being a *collision attack*:

```
def collision_attack(num_bits):
    attempts = 0
    hashes = {}

    while True:
        hash = my_sha_1(random_word(), num_bits)

        if hashes.contains(hash):
            return attempts
        else:
            hashes.add(hash)
            attempts += 1
```

The *collision attack* is defined as finding any two messages that compute the same hash: $H(m_1) = H(m_2) \mid m_1 != m_2$. This is also known as the birthday attack, where the general approximate probability of finding two random hash collisions over a domain of fixed permutations of n bits is: $2^{n/2}$ (Girault, Cohen, & Campana, 1988). This attack tests the hashing algorithm's resistance to strong collisions.

The second attack type of this experiment is a *pre-image attack*:

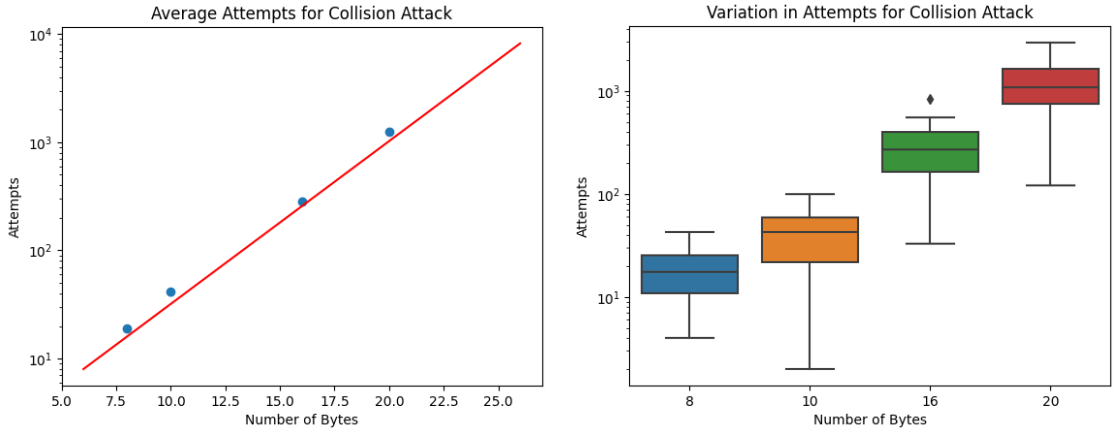
```
def pre_image_attack(num_bits):
    attempts = 0
    hash = my_sha_1(random_word(), num_bits)

    while True:
        attempts += 1

        new_hash = my_sha_1(random_word(), num_bits)

        if hash == new_hash:
            return attempts
```

More specifically, this attack follows the design of a *first pre-image attack*, where the attack must find a message that creates an identical hash to a given digest: $H(m) = d \mid d$ *is given*. This type of *pre-image attack* tests the hash algorithm's one-way ability, meaning that it is computationally difficult to find a new message that computes the same hash of a certain digest. The probability of this attack can be generalized to an average number of attempts: 2^n for SHA-1 (where n represents the number of bits in the hash) (Preshing, 2011).

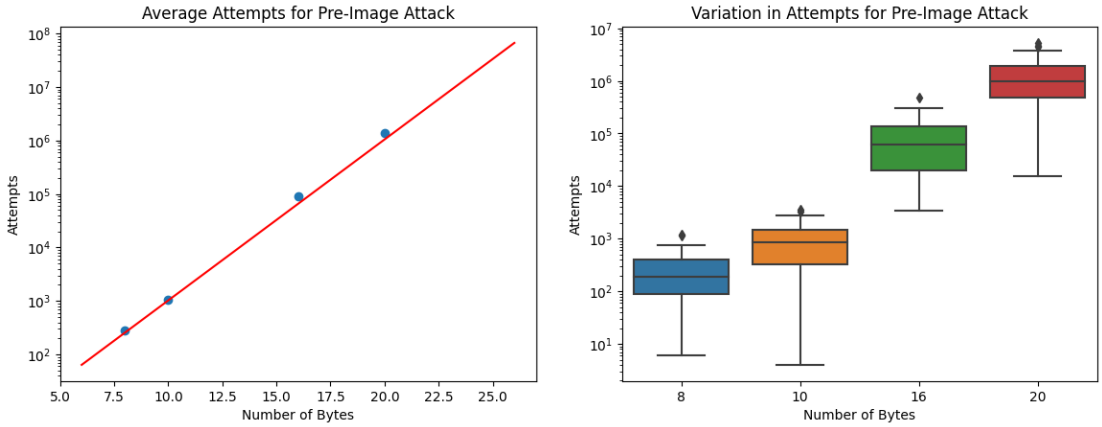
To compare the SHA-1 attack results to the theoretical results, we run the attack algorithms with 4 different bit sizes ranging from 8 to 20 (*8, 10, 16, 20*) and perform 100 attacks for each bite size (for a total of 400 *collision attacks* and 400 *pre-image attacks*). This should give us enough data to accurately calculate the average attempts to complete each attack.

Results

The average attempts for the *collision* and *pre-image attacks* are as follows: (As represented in a logarithmic graph. The blue dots showing the average number of attempts for each bit size, and the red line representing the theoretical probability of $2^{n/2}$ and 2^n respectively.)



Looking at the plotted results, we see that the *collision attack* closely follows the expected results. The *collision attack* results are always a little above the expected line due to the prediction equation of $2^{n/2}$ being a lower bound. This is because the possible number of attempts ranges from zero to infinity. The space below the expected number of attempts is thus capped at zero, while the space above the predicted line in infinite. Hence, we expect (and see) that the actual results are a little above the predicted number. This is also reflected in the boxplot, as the median bar is in or above the middle of the colored section (representing Q_1 through Q_3).



When looking at the results for the *pre-image* attack we can see that data closely follows the predicted line of: 2^n. This is because the prediction equation for *pre-image attacks* (2^n) is not a lower bound as stated by Preshing (2011). Thus, we expect (as see) that our actual results closely adhere to the red prediction line. This is also reflected in the boxplot, where the median is closer to the middle of the colored sections (Q_1-Q_3).

Additionally, Anthony Glad (a member of the CS 465 course) has looked over this report as an external reviewer, and has confirmed these findings.

Conclusion

In conclusion, we see that the *collision attack* is much more scalable than the *pre-image attack* because of its increased probability of success ($2^{n/2} < 2^n$). We also compared our experiments (over a wide domain of bit sizes and over many iterations) to the theoretical probability equations, finding that the experiments closely follow the theoretical number of attempts. We also explained why the results are slightly above the predicted number of attempts for the *collision attack*. Thus, we conclude that the efficacies of *collision* and *pre-image attacks* over SHA-1 are well represented by their probability equations.

References

Girault, M., Cohen, R., & Campana, M. (1988). A generalized birthday attack. Paper presented at the *Workshop on the Theory and Application of of Cryptographic Techniques*, 129-156.

Preshing, J. (2011). Hash collision probabilities. Retrieved from https://preshing.com/20110504/hash-collision-probabilities/