Using Neural Network Chains to Reverse the Avalanche Effect in Reduced-Round SHA-1

With the advent of the information age, there is a rising need for digital security. Personal information, medical data, passwords, and IoT devices are just a few examples of critical information systems that need to be protected. Cryptographic hashes are essential to this goal. These deterministic functions map a variable-length message (the plaintext) to a fixed-length hash value (the ciphertext) (Fig. 1), but in a manner that is difficult to reverse and appears random. These are used to securely encrypt information. For example, when a user enters a new password to a website, that password is first hashed before being saved in the database. When that user wants to log in again, the entered password is also hashed and compared to the saved hash. Even if a hacker breaks into the database, the original password is not discoverable, so the user's account is safe. Hence, it is critical that cryptographic hashes continue to be improved upon as technologies advance to ensure a safer Internet.

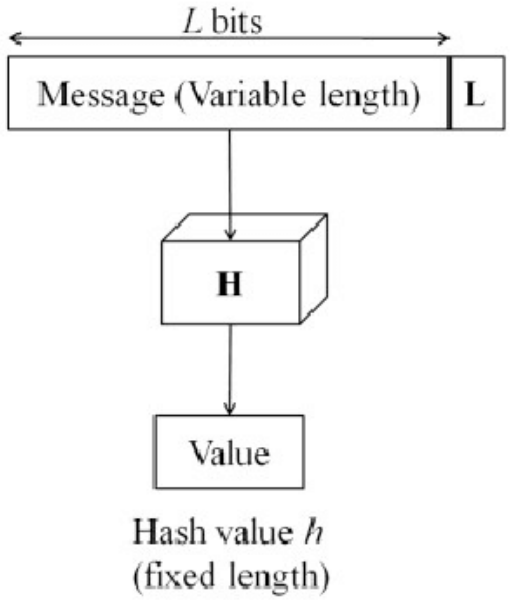

Fig. 1. Structure of a hash function (Sharma & Bhatt, 2018)

Past Work

Hashes can only be improved by first finding their weak points. As an analogy, a leaky pipe can only be fixed once the hole is identified. This area of study, analyzing and exploiting cryptographic systems to deduce secret aspects of those systems, is called cryptanalysis. One subsection of cryptanalysis is neuro-cryptanalysis, or cryptanalysis using machine learning. Although this field is relatively small, there have been some advances made over the past decade. First, in his seminal 1993 paper, Rivest described the potential links between machine learning and cryptanalysis, going so far as to dub them "sister fields." In 2012, Alani conducted a known-plaintext attack using a cascading neural

network trained on plaintext-ciphertext pairs to predict the plaintext of the DES and Triple-DES ciphers based on the ciphertext without knowing the secret key. This attack successfully reduced both the time and data needed to conduct a known-plaintext attack on these ciphers. In 2017, Greydanus used recurrent neural networks to successfully, albeit inefficiently, reverse polyalphabetic ciphers, specifically the Vignere, Autokey, and Enigma ciphers, which are much weaker than modern cryptographic hashes. In 2019, Goncharov trained a standard feed-forward neural network on plaintext-ciphertext pairs from several common cryptographic hashes while making use of fuzzy logic, which extends boolean values to all values between 0 and 1. However, they only achieved success in extremely low-round versions of the hashes. Also in 2019, Gohr presented a new strategy using deep neural networks that distinguished between random data and outputs produced by a reduced-round Speck cipher given an input difference, greatly aiding in differential cryptanalysis; this neural distinguisher surpassed state-of-the-art techniques in the amount of rounds it successfully attacked. In 2020, So successfully launched a known-plaintext attack on SDES, Simon, and Speck by restricting the keyspace to 64 ASCII characters, skewing the key bit distribution. In 2021, Benamira et al. built upon this work by translating the deep neural network into pure cryptanalysis, creating an optimized version of the machine learning model, and further improving it through modifications such as adding batches. In the same year, Liu et al. used deep neural networks to launch a preimage attack against the reduced-round Xoodyak hash by training them on plaintext-ciphertext pairs, achieving limited success in one-round Xoodyak. Overall, neuro-cryptanalytic techniques have achieved widely varying levels of success in the past. However, most of this research used neural networks without considering the hash's internal structure, which may make their approaches less effective.

Research

Most past research has failed to consider the algorithms' architecture, instead using neural networks as black-boxes to try to cover the entire function at once. By breaking the algorithm into its

components and linking together several neural networks to analyze each, greater efficiency and accuracy could potentially be achieved. Therefore, the goal of this project is to construct a system to apply neural network chains to conduct a preimage attack on reduced-round SHA-1. A preimage attack is the quintessential cryptanalytic attack, where the attacker finds the plaintext based on the ciphertext of a hash. The chosen attack model is Secure Hashing Algorithm 1, or SHA-1, due to its suitable architecture and the extensive research and documentation on it. A successful attack would serve as a proof-of-concept for a new approach to cryptanalysis. This project also aims to explore potential connections between neural networks and other existing attack approaches such as meet-in-the-middle attacks, restricted-key-space attacks, and SAT-solver-based attacks.

Researchable Question

Can neural network chains be used to conduct a preimage attack?

Objective

Obj. 1: The first objective of this project is to train individual neural networks to reverse rounds of SHA-1 with adequate accuracy, preferably above 90%.

Obj. 2: The second objective of this project is to link together the neural networks constructed from Objective 1, where the plaintext fragments extracted from the chain are correct with nontrivial accuracy.

Obj. 3: The third objective of this project is to extend neural network chain length until trivial accuracy is achieved and apply possible extensions such as meet-in-the-middle attack techniques.

Hypothesis

Linking together neural networks of various architectures that are individually trained on data from each round of the cryptographic hash SHA-1 will reduce the complexity each network needs to model, allowing reconstruction of the internal state history and extraction of the preimage.