CSE441: Project Proposal
Cryptanalysis of Hash Functions

Cole Johnson: cole.johnson.@student.nmt.edu
John Runyon: john.runyon@student.nmt.edu

September 17, 2024

Introduction and Outline of Project

The Basics of Hash Functions

Hashing, as an overall process, is a function that maps plaintext data of any length into a fixed-length ciphertext output–often called a digest. Hash functions, unlike encryption, destroy information encoded in the plaintext, which means the function is one-way and cannot be reversed to obtain the plaintext again.

Hash functions are a widely used type of cryptographic algorithm. They can be used for a variety of purposes, such as data integrity verification, password storage, and digital fingerprint/signatures, and data indexing (often called hash-tables). Since hash functions serve vital purposes in modern cryptography and computer science, knowing the important mathematical properties of a hash function (and how these can be implemented in programs) is critical to understanding their function.

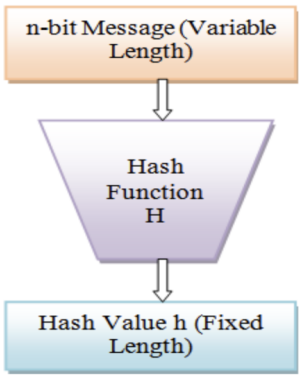

Figure 1: Basic diagram of a hash function

Once the function and real-world implementation of a hash function is understood, it becomes clear why certain mathemati-

cal properties, such as determinism, pre-image resistance, and collision resistance are crucial. These properties ensure that hash functions can efficiently convert data of any size into a fixed-length output while preventing malicious actors from reversing or tampering with the data. The fixed length output is a major factor of any hashing algorithm - meaning that a user can input one character or one-hundred characters, and still receive a 256-bit hash as is the case with SHA-256.

Outline of Project

Here is the sections of the project, along with a small description of what each of the sections will cover:

(a.) Introduction to Hash Functions

- (a) Basics of Hash Functions (including code examples, and diagrams)
- (b) Legacy/Classic and Modern Hash Functions and their Applications
- (c) Hash Functions Examples (MD5, SHA-256, Tornado)

(b.) Properties of Hash Functions

- (a) One-way Function (Pre-Image Resistance)
- (b) Target Collision Resistance (2nd Pre-Image Resistance)
- (c) Deterministic
- (d) Avalanche Effect
- (e) Computational Speed

(c.) Cryptanalysis and Attacks on Cryptographic Hash Functions

- (a) Brute-Force Attacks
- (b) One-way Function Inversion
- (c) 2nd Pre-Image Resistance Attack
- (d) Collision Attack

Experimentation

For our experimentation our group is wanting to look into the varying overhead required by different cryptographic algorithms such as MD5, SHA-1, and SHA-256.

This could include using local machines for testing and online research to find and compare the available data on the performance of different hash functions in terms of speed and the use of resources. Our reasoning and conclusion of this experimentation would result in a conversation on the trade-offs between security and speed and the different use cases for each.

Conclusion

Our project will explore the process of cryptoanalysis with various hash functions as mentioned above. By analyzing the performance, and security of varying hash functions we aim to better understand the trade-offs that exists amongst these algorithms. Our testing of these functions will give us a real world example of the hashes and the implications especially in terms of the overhead involved.

Hash functions are critical to the modern internet and computer infrastructure, so understanding their uses is not only important, but critical for security and ensuring data integrity (with uses of older hash functions as checksums). Our project will end with a review of the current state of hashing and what the future might look like in the world of post-quantum cryptography (PQC).