Cryptographic hash functions

Diego F. Aranha

Institute of Computing
UNICAMP

Introduction

Objectives:

- Study properties and constructions for cryptographic hashing.

Introduction

Objectives:

- Study properties and constructions for cryptographic hashing.

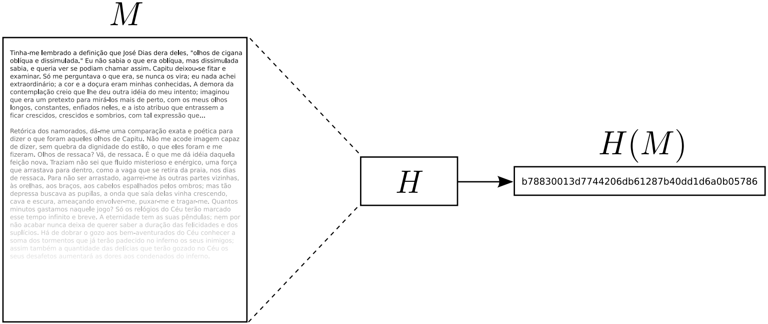
Hidden intentions:

- Simultaneously infer the limitations of cryptographic hash functions.

Cryptographic hash functions

Informal definition

Cryptographic hash functions are employed to produce a short descriptor of a message. Informally, this descriptor is analogous to a fingerprint for human identification.



Cryptographic hash functions

Formal definition

A **cryptographic hash function** maps messages from a set $\mathcal{X}$ to hash values or authenticators in a set $\mathcal{Y}$. In this first case, it is denoted by $h : \mathcal{X} \rightarrow \mathcal{Y}$. In the second, it is parameterized by a key $K \in \mathcal{K}$ and represented by $h_K : \mathcal{X} \rightarrow \mathcal{Y}$. If $\mathcal{X}$ is finite h is also called a **compression function**.

Many different applications:

- Password storage (store $h(s)$ instead of s).
- Key derivation ($k = h(g^{xy} \bmod p)$, $k_i = h(k_{i-1})$).
- Integrity verification ($y = h(x)$).
- Digital signatures (sign $h(m)$ instead of just m).
- Message Authentication Codes (MACs) ($y = h_K(x)$).


HASH ALL THE THINGS!
memegenerator.net

Properties of hash functions

- **Preimage resistance**: Given hash y, it should be computationally infeasible to find x such that $y = h(x)$.

- **Second preimage resistance**: Given hash y and a message x such that $y = h(x)$, it should be computationally infeasible find $x' \neq x$ such that $h(x') = h(x) = y$.

- **Collision resistance**: It should be computationally infeasible to find x, x' such that $h(x) = h(x')$.

Important: Each property implies the previous one (in the first case, conditionally).

Properties of hash functions

Collision from second preimage

1. Choose random x.
2. Compute $y = H(x)$.
3. Obtain second preimage $x' \neq x$ such that $H(x') = H(x) = y$.
4. Return collision (x, x').

Second preimage from first preimage

1. Compute $y = H(x)$.
2. Invert $x' = H^{-1}(y)$ until you obtain $x' \neq x$.
3. Return collision (x, x').

Important: If $|\mathcal{X}| \geq 2|\mathcal{Y}|$, not possible to obtain collision resistance if S is not resistant to both first and second preimages!

Properties of hash functions

From the reductions:

- Collision resistance implies second preimage resistance.
- If $|\mathcal{X}| \geq 2|\mathcal{Y}|$, collision resistance implies preimage resistance.
- Finding collisions has no impact to first and second preimages.
- Not possible to find first or second preimages without affecting collision resistance.

Hash functions design

- **Merkle-Damgård paradigm**: MD4, MD5, SHA-1, SHA-2.
- Block cipher-based: Matyas-Meyer-Oseas, David-Meyer.
- New paradigms: Sponge (SHA3/Keccak).
- Number theory: VHS (integer factoring), ECOH (elliptic curves).

Random Oracle Model (ROM)

Definition

The **Random Oracle Model** is a mathematical model of an *ideal* hash function: the function is chosen randomly from all such functions $f : \mathcal{X} \rightarrow \mathcal{Y}$ and represented by an oracle. Because the formula or algorithm are unknown, the only way to compute the hash function is to sample the oracle.

Random Oracle Model (ROM)

Definition

The **Random Oracle Model** is a mathematical model of an *ideal* hash function: the function is chosen randomly from all such functions $f : \mathcal{X} \rightarrow \mathcal{Y}$ and represented by an oracle. Because the formula or algorithm are unknown, the only way to compute the hash function is to sample the oracle.

Corollary: Outputs are independently and uniformly distributed!

Random Oracle Model (ROM)

Definition

The **Random Oracle Model** is a mathematical model of an *ideal* hash function: the function is chosen randomly from all such functions $f : \mathcal{X} \rightarrow \mathcal{Y}$ and represented by an oracle. Because the formula or algorithm are unknown, the only way to compute the hash function is to sample the oracle.

Corollary: Outputs are independently and uniformly distributed!

Advantages: Models the security requirements of hash functions and allows reducing the security of protocols to oracle properties.

Random Oracle Model (ROM)

Definition

The **Random Oracle Model** is a mathematical model of an *ideal* hash function: the function is chosen randomly from all such functions $f : \mathcal{X} \rightarrow \mathcal{Y}$ and represented by an oracle. Because the formula or algorithm are unknown, the only way to compute the hash function is to sample the oracle.

Corollary: Outputs are independently and uniformly distributed!

Advantages: Models the security requirements of hash functions and allows reducing the security of protocols to oracle properties.

Disadvantages: Real hash functions are not ideal!

Birthday paradox

It is a classic problem that demonstrates how counter-intuitive results in probability can be to the human brain.

Definition

What is the minimum value k such that the probability of two persons in a room with k people share their birthdays is higher than 50%?

Birthday paradox

Let $p'(n), n \leq 365$ the probability that all birthdays are different:

$$p'(n) = 1 \cdot \left(1 - \frac{1}{365}\right) \cdot \left(1 - \frac{2}{365}\right) \cdot \ldots \cdot \left(1 - \frac{n-1}{365}\right) = \frac{365!}{(365-n)!365^n}$$

We have that $p(n) = 1 - p'(n)$. Thus, $p(n) > 0.5$ if $n \geq 23$ and $p(n) = 1$ if $n \geq 100$.

Important: With only $k = 23$ people, the probability that two of them share birthdays is already over 50%!

Birthday paradox

Let $p'(n), n \leq 365$ the probability that all birthdays are different:

$$p'(n) = 1 \cdot \left(1 - \frac{1}{365}\right) \cdot \left(1 - \frac{2}{365}\right) \cdot \ldots \cdot \left(1 - \frac{n-1}{365}\right) = \frac{365!}{(365-n)!365^n}$$

We have that $p(n) = 1 - p'(n)$. Thus, $p(n) > 0.5$ if $n \geq 23$ and $p(n) = 1$ if $n \geq 100$.

Important: With only $k = 23$ people, the probability that two of them share birthdays is already over 50%!

Important: Do not confuse with the much probability of another person in the room share a fixed birthday $q(n) = 1 - \left(\frac{364}{365}\right)^n$.

Birthday attack

Generalizing to hash functions where $|\mathcal{Y}| = M$, the probability of finding collisions after n random samples is:

$$p(n) = 1 - \left(1 - \frac{1}{M}\right) \cdot \left(1 - \frac{2}{M}\right) \cdot \ldots \cdot \left(1 - \frac{n-1}{M}\right) \approx 1 - e^{-n(n-1)/(2M)}$$

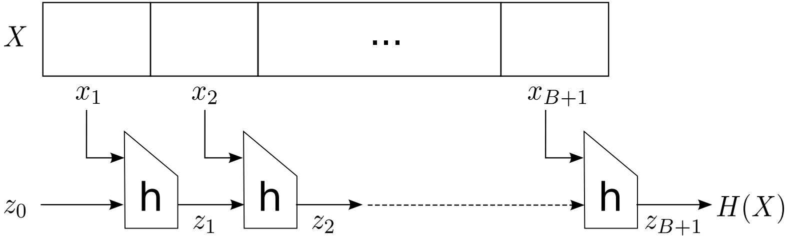
Replacing $p(n) = \frac{1}{2}$ and solving for n, we have that $n \approx 1.17\sqrt{M}$. In other words, sampling more than $\sqrt{M}$ elements should produce a collision with probability of 50%.

Important: That is why hash functions with output length of m *bits* offer security of only $\frac{m}{2}$ *bits*!

Iterated hash functions (Merkle-Damgård)

Definition

It is a technique that allows constructing a hash function with infinite domain $H : \{0,1\}^* \rightarrow \{0,1\}^m$ through consecutive applications of a **compression function** $h : \{0,1\}^{m+t} \rightarrow \{0,1\}^m$. **Padding** is needed for adding block x_{B+1} to an input x with B blocks.



Important: Collision resistance for S is given by collision resistance for h.

SHA-1 hash function

Definition

It is a cryptographic hash function $H : \{0,1\}^{2^{64}} \rightarrow \{0,1\}^{160}$ following the Merkle-Damgård paradigm.

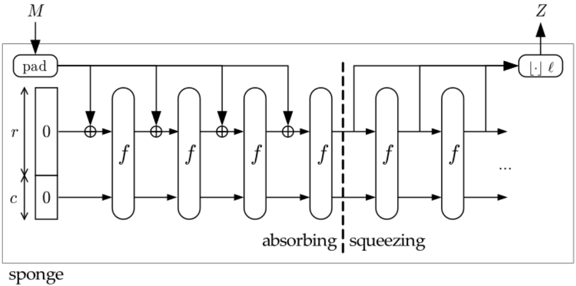
Brief history:

- Proposed by NIST in 1993.
- It is an improvement over SHA-0 (collision in 2^{61} operations).
- It is an 80-round iterated hash function with compression function $h : \{0,1\}^{512} \rightarrow \{0,1\}^{160}$.
- After attacks, SHA-2 and SHA-3 became standard.
- Security estimated in 60 *bits*.

Iterated hash functions (Sponge)

Definition

The **sponge construction** is a mode of operation based on a *fixed-length permutation* and a *padding rule*, which builds a function mapping variable-length input to variable-length output. A sponge function is a generalization of both hash functions, which have a fixed output length, and stream ciphers, which have a fixed input length.



Important: Collision resistance depends on **internal state size**!