Memoryless Near-Collisions via Coding Theory

Mario Lamberger   Florian Mendel
Vincent Rijmen   Koen Simoens

Institute for Applied Information Processing and Communications (IAIK)
Graz University of Technology
Inffeldgasse 16a, A-8010 Graz, Austria

`mario.lamberger@iaik.tugraz.at`

Memoryless Collision

- I guess we heard about the birthday paradox
  - For an n-bit hash function, we need $2^{n/2}$ hash calls and a list of the same size
- Using a lot of memory sucks, so we
- implement it using a cycle finding method
  - Floyd
  - Brent
  - ...

Now what about near-collisions

Near-Collision Resistance - HAC

It should be hard to find any two inputs m, m^* such that $H(m)$ and $H(m^*)$ differ in only a small number of bits:

$$\mathrm{d}(H(m), H(m^*)) \leq \epsilon.$$

- This includes collisions $\Rightarrow$ easier!
- What should a “near”-cycle be?

A possible solution

- π: Linear projection map that sets ϵ bits to 0

A possible solution

- π: Linear projection map that sets ϵ bits to 0
- Then, a collision for $\pi \circ H$ results in a near-collision for H

A possible solution

- π: Linear projection map that sets ϵ bits to 0
- Then, a collision for $\pi \circ H$ results in a near-collision for H
- Improves the performance by $2^{\epsilon/2}$
- Drawback: finds only a fraction of all ϵ-near-collisions

$$\frac{2^{\epsilon}}{\sum_{i=0}^{\epsilon} \binom{n}{i}}.$$

A possible solution

- π: Linear projection map that sets ϵ bits to 0
- Then, a collision for $\pi \circ H$ results in a near-collision for H
- Improves the performance by $2^{\epsilon/2}$
- Drawback: finds only a fraction of all ϵ-near-collisions

$$\frac{2^{\epsilon}}{\sum_{i=0}^{\epsilon} \binom{n}{i}}.$$

- Ideally, we would like to have a map g which gives a one-to-one correspondence between ϵ-near-collisions ($\epsilon \geq 1$) for H and collisions for $g \circ H$

Our idea

- Let H be a hash function of output size n.

Our idea

- Let H be a hash function of output size n.
- Let $\mathcal{C} \subseteq \mathbb{Z}_2^n$ be a code of the same length n, size K and covering radius $\rho(\mathcal{C})$ and assume there exists an efficiently computable map g that maps every $x \in \mathbb{Z}_2^n$ to a codeword at distance $\rho(\mathcal{C})$ or less

Our idea

- Let H be a hash function of output size n.
- Let $\mathcal{C} \subseteq \mathbb{Z}_2^n$ be a code of the same length n, size K and covering radius $\rho(\mathcal{C})$ and assume there exists an efficiently computable map g that maps every $x \in \mathbb{Z}_2^n$ to a codeword at distance $\rho(\mathcal{C})$ or less
- Then, we can find $2\rho(\mathcal{C})$-near-collisions for H with a complexity of about $\sqrt{K}$ and with virtually no memory requirements

Our idea

- Let H be a hash function of output size n.
- Let $\mathcal{C} \subseteq \mathbb{Z}_2^n$ be a code of the same length n, size K and covering radius $\rho(\mathcal{C})$ and assume there exists an efficiently computable map g that maps every $x \in \mathbb{Z}_2^n$ to a codeword at distance $\rho(\mathcal{C})$ or less
- Then, we can find $2\rho(\mathcal{C})$-near-collisions for H with a complexity of about $\sqrt{K}$ and with virtually no memory requirements
- If decoding is efficient, use this as g

Our idea

- Let H be a hash function of output size n.
- Let $\mathcal{C} \subseteq \mathbb{Z}_2^n$ be a code of the same length n, size K and covering radius $\rho(\mathcal{C})$ and assume there exists an efficiently computable map g that maps every $x \in \mathbb{Z}_2^n$ to a codeword at distance $\rho(\mathcal{C})$ or less
- Then, we can find $2\rho(\mathcal{C})$-near-collisions for H with a complexity of about $\sqrt{K}$ and with virtually no memory requirements
- If decoding is efficient, use this as g
- Size $K \to$ sphere covering bound

Our proposed construction

- For given n and ρ we considered direct sums of Hamming codes and trivial codes

$$\mathcal{C} = \bigoplus_{i \geq 1} d_i \mathcal{H}_i \oplus \mathbb{Z}_2^{r(n,\rho)}$$

Our proposed construction

- For given n and ρ we considered direct sums of Hamming codes and trivial codes

$$\mathcal{C} = \bigoplus_{i \geq 1} d_i \mathcal{H}_i \oplus \mathbb{Z}_2^{r(n,\rho)}$$

- Easy to decode

Our proposed construction

- For given n and ρ we considered direct sums of Hamming codes and trivial codes

$$\mathcal{C} = \bigoplus_{i \geq 1} d_i \mathcal{H}_i \oplus \mathbb{Z}_2^{r(n,\rho)}$$

- Easy to decode
- Gives rise to an interesting digit problem
  - $\sum_{i \geq 1} d_i N_i \leq n, \quad N_i = 2^i - 1, d_i \in \{0, \ldots, \rho\}$
  - $\sum_{i \geq 1} d_i = \rho$
  - $\sum_{i \geq 1} d_i \cdot i$ should be maximal

Our proposed construction

- For given n and ρ we considered direct sums of Hamming codes and trivial codes

$$\mathcal{C} = \bigoplus_{i \geq 1} d_i \mathcal{H}_i \oplus \mathbb{Z}_2^{r(n,\rho)}$$

- Easy to decode
- Gives rise to an interesting digit problem
  - $\sum_{i \geq 1} d_i N_i \leq n, \quad N_i = 2^i - 1, d_i \in \{0, \ldots, \rho\}$
  - $\sum_{i \geq 1} d_i = \rho$
  - $\sum_{i \geq 1} d_i \cdot i$ should be maximal

- Demonstrated the approach on the SHA-3 candidate TIB-3

Thank you for your attention!