

# A função *hash* criptográfica SHA-3

Gustavo Zambonin\*

Segurança em Computação (UFSC – INE5429)

## 1 Definições

- Uma *função hash criptográfica*, ou função de resumo criptográfica (futuramente denotada por  $h$ ), é um algoritmo matemático que mapeia uma quantidade de bytes qualquer<sup>1</sup> para uma palavra de tamanho fixo, ou seja,  $h : \{0, 1\}^* \rightarrow \{0, 1\}^n$ ,  $n \in \mathbb{N}$ .

Para que seja resistente a diversos tipos de criptoanálise, uma função  $h : X \rightarrow Y$  deve respeitar algumas propriedades:

- i. *Resistência à pré-imagem*: Para um resumo  $M' \in Y$ , é computacionalmente impraticável<sup>2</sup> encontrar a mensagem  $M \in X$  tal que  $h(M) = M'$ . Uma função matemática com esta propriedade é chamada de unidirecional.
- ii. *Resistência à segunda pré-imagem*: Para uma mensagem  $M_0 \in X$ , é computacionalmente impraticável encontrar uma segunda mensagem  $M_1 \in X$  tal que  $M_0 \neq M_1$  e  $h(M_0) = h(M_1)$ .
- iii. *Resistência à colisão*: Para duas mensagens  $M_0, M_1 \in X$ , é computacionalmente impraticável encontrar  $M_0 \neq M_1$  e  $h(M_0) = h(M_1)$ .

É importante notar que, embora as definições sejam extremamente parecidas, resistência à segunda pré-imagem e resistência à colisão são conceitos diferentes; um atacante não consegue escolher a primeira mensagem caso queira atacar a resistência à segunda pré-imagem; para a resistência à colisão, o atacante pode escolher livremente o par de mensagens.

- Algumas aplicações destas funções são enumeradas abaixo:
  - Podem ser utilizadas para verificar a integridade da mensagem, comparando resumos criptográficos calculados antes e depois da transmissão de mensagem e/ou arquivos.
  - Para evitar o armazenamento de senhas em texto claro, é possível armazenar apenas o resumo criptográfico de cada senha e compará-lo na autenticação do usuário.
  - Resumos criptográficos são comumente descritos como identificadores únicos seguros para um arquivo ou informação digital (por exemplo, *commits* em um sistema de controle de versão).
- O padrão SHA-3, descrito pelo documento FIPS 202 [3], é baseado em uma instância da família KECCAK de permutações matemáticas, selecionada pelo NIST (*National Institute of Standards and Technology*) e especificada neste documento.

## 2 O algoritmo SHA-3

- KECCAK é uma família de funções esponja. Este tipo de função é uma generalização do conceito da função de resumo criptográfica com saída infinita. Após a aplicação de uma função de preenchimento (*padding*) à mensagem  $M$ , a função esponja tem duas fases: a fase de absorção (*absorbing*), responsável por intercalar blocos de  $M$  com aplicações de uma função de permutação  $f$ , de modo iterativo; e a fase de compressão (*squeezing*), onde os blocos de saída, intercalados novamente pela permutação  $f$ , são concatenados para gerar uma palavra com um número de bits configurável pelo usuário. Esse processo pode ser observado na figura 1.
- A permutação  $f$  é descrita como uma sequência de operações num estado  $A$ , que é um vetor de elementos tridimensional em  $GF(2)$ , chamado de  $A$ .  $f$  é uma permutação iterativa, consistindo de uma sequência de rodadas  $R$ . Uma rodada consiste da composição de cinco etapas, ilustradas em 2:

$$R = \iota \circ \chi \circ \pi \circ \rho \circ \theta$$

\*gustavo.zambonin@grad.ufsc.br — todos os algoritmos utilizados podem ser encontrados também [neste repositório](#).

<sup>1</sup>algumas funções desse tipo têm limites quanto ao tamanho da entrada, embora estes sejam extremamente grandes.

<sup>2</sup>o tempo ou recursos gastos para esta computação excedem a validade ou utilidade da informação desejada.

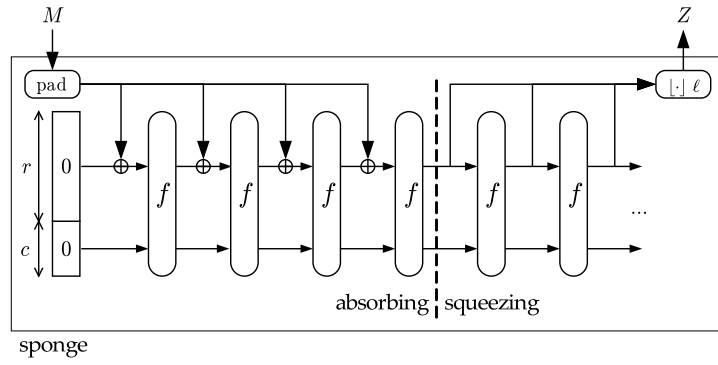


Figura 1: Uma construção esponja. Imagem retirada de [1].

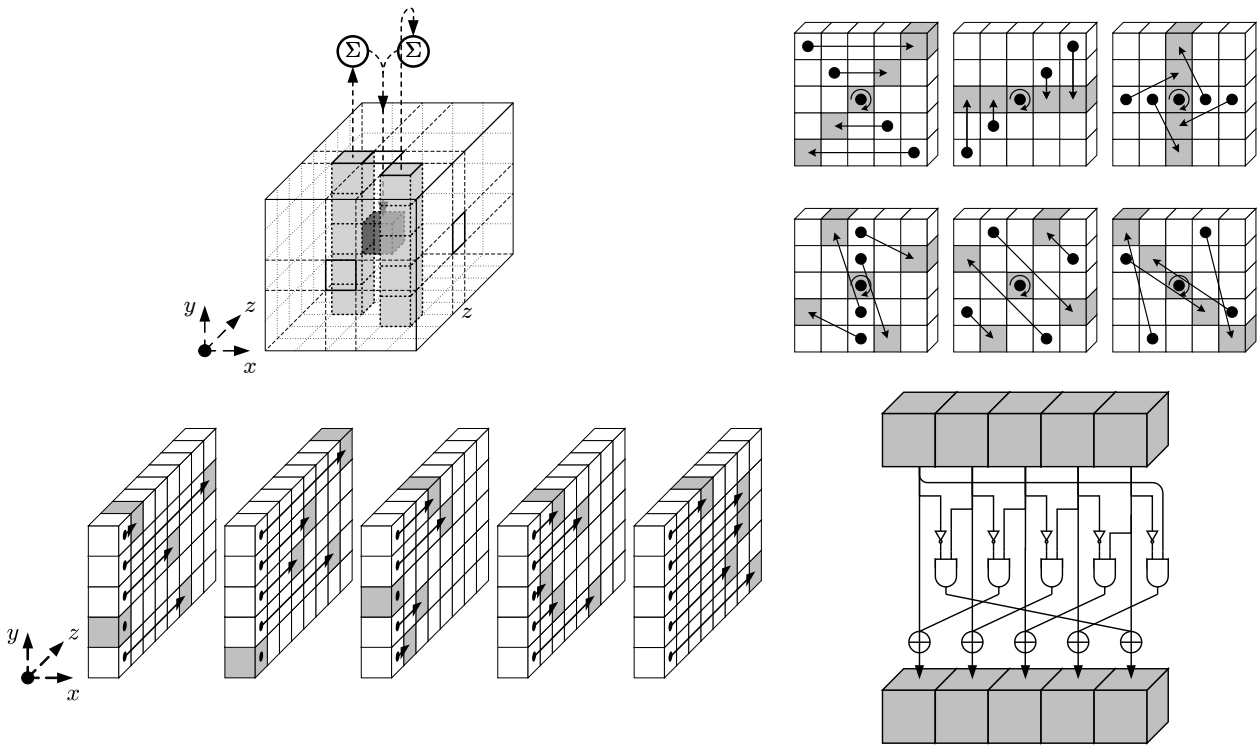


Figura 2: Da esquerda para a direita e de cima para baixo, as etapas  $\theta$ ,  $\pi$ ,  $\rho$  e  $\chi$ . Imagens retiradas de [2].

- A etapa  $\theta$  faz a soma XOR de um elemento de  $A$  e todos os elementos das colunas adjacentes indicadas.
- A etapa  $\rho$  dispersa os elementos entre cortes transversais verticais de  $A$ .
- A etapa  $\pi$  rearranja as posições de elementos em cortes transversais horizontais de  $A$ .
- A etapa  $\chi$  tem como efeito fazer a soma XOR de cada bit em uma linha, de acordo com uma função não-linear de dois outros bits adjacentes.
- A etapa  $\iota$  é utilizada para quebrar a simetria das operações acima, e sem esta etapa, todas as rodadas teriam a mesma saída. A soma XOR de alguns bits do estado  $A$  é feita com um bit específico de uma sequência gerada por um LFSR<sup>3</sup>, alimentado pelo índice da rodada atual.

## Referências

- [1] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche. Cryptographic sponge functions, January 2011.
- [2] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche. The KECCAK reference, January 2011. <http://keccak.nokeon.org/>.
- [3] Morris J. Dworkin. SHA-3 standard: Permutation-based hash and extendable-output functions. Technical report, July 2015.

---

<sup>3</sup>*linear-feedback shift register*, um tipo de gerador de sequências pseudoaleatórias.