Função de Hash Criptográfica SHA-3

Quênio César Machado dos Santos (14100868)

INE5429 - Segurança em Computadores

Florianópolis, 14/06/2016

Sumário

1 Função Hash Criptográfica

Uma função *hash* é uma função que aceita um bloco de dados de tamanho variável como entrada e produz um valor de tamanho fixo como saída, chamado de valor *hash*. Esta função tem a forma:

$$h = H(M)$$

Onde:

- h é o valor *hash* de tamanho fixo gerado pela função *hash*.
- H é função *hash* que gerou o valor h.
- M é o valor de entrada de tamanho variável.

Espera-se que uma função *hash* produza valores h que são uniformemente distribuídos no contra-domínio e que são aparentemente aleatórios, ou seja, a mudança de apenas um *bit* em M causará uma mudança do valor h. Por esta característica, as funções *hash* são muito utilizadas para verificar se um determinado bloco de dados foi indevidamente alterado.

As funções *hash* apropriadas para o uso em segurança de computadores são chamadas de "função *hash* criptográfica". Este tipo de função *hash* é implementada por um algoritmo que torna inviável computacionalmente encontrar:

- um valor M dado um determinado valor h:

$$M \mid H(M) = h$$

- dois valores M_1 e M_2 que resultem no mesmo valor h:

$$(M_1, M_2) \mid H(M_1) = H(M_2)$$

Os principais casos de uso de funções *hash* criptográficas são:

- *Autenticação de Mensagens*: é um serviço de segurança onde é possível verificar que uma mensagem não foi alterada durante sua transmissão e que é proveniente do devido remetente.
- *Assinatura Digital*: é um serviço de segurança que permite a uma entidade assinar digitalmente um documento ou mensagem.
- *Arquivo de Senhas de Uma Via*: é uma forma de armazenar senhas usando o valor *hash* da senha, permitindo sua posterior verificação sem a necessidade de armazenar a senha em claro, cifrá-la ou decifrá-la.
- *Detecção de Perpetração ou Infeção de Sistemas*: é um serviço de segurança em que é possível determinar se arquivos de um sistema foram alterados por terceiros sem a autorização dos usuários do sistema.

1.1 Propriedades

Como observado na seção anterior, uma função *hash* criptográfica precisa ter certas propriedades para permitir seu uso em segurança de computadores. Nas seções a seguir estão destacadas algumas dessas propriedades.

Antes, defini-se dois termos usados a seguir:

- *Pré-Imagem*: um valor M do domínio de uma função *hash* dada pela fórmula $h = H(M)$ é denominado de "pré-imagem" do valor h.
- *Colisão*: para cada valor h de tamanho n *bits* existe necessariamente mais de uma pré-imagem correspondente de tamanho m *bits* se $m > n$, ou seja, existe uma "colisão".

O número de pré-imagens de m *bits* para cada valor h de n *bits* é calculado pela formula: $2^{\frac{m}{n}}$. Se permitimos um tamanho em *bits* arbitrariamente longo para as pré-imagens, isto aumentará ainda mais a probabilidade de colisão durante o uso de uma função *hash*. Entretanto, os riscos de segurança são minimizados se a função de *hash* criptográfica oferecer as propriedades descritas nas próximas seções.

1.1.1 Resistente a Pré-Imagem

Uma função *hash* criptográfica é resistente a pré-imagem quando esta é uma função de uma via. Ou seja, embora seja computacionalmente fácil gerar um valor h a partir de uma pré-imagem M usando a função de *hash*, é computacionalmente inviável gerar uma pré-imagem a partir do valor h.

Se uma função *hash* não for resistente à pré-imagem, é possível atacar uma mensagem autenticada M para descobrir o valor secreta S usada na mensagem, permitindo assim ao perpetrante enviar uma outra mensagem M_2 ao destinatário no lugar do remetente sem que o destinatário perceba a violação da comunicação. O ataque ocorre da seguinte forma:

- O perpetrante tem conhecimento do algoritmo de *hash* $h = H(M)$ usado na comunicação entre as partes.
- Ao escutar a comunicação, o perpetrante descobre qual é a mensagem M e o valor de *hash* h.
- Visto que a inversão da função de *hash* é computacionalmente fácil, o perpetrante calcula $H^{-1}(h)$.
- Como $H^{-1}(h) = S \parallel M$, o perpetrante descobre S.

Desta forma, o perpetrante pode utilizar a chave secreta S no envio de uma mensagem M_2 para o destinatário sem que este perceba a violação.

1.1.2 Resistente a Segunda Pré-Imagem

Uma função *hash* criptográfica é resistente a segunda pré-imagem quando esta função torna inviável computacionalmente encontrar uma pré-imagem alternativa que gera o mesmo valor h da primeira pré-imagem.

Se uma função de *hash* não for resistente a segunda pré-imagem, um perpetrante conseguirá substituir uma mensagem que utiliza um determinado valor de *hash*, mesmo que a função de *hash* seja de uma via, ou seja, resistente a pré-imagem.

1.1.3 Resistente a Colisão

Uma função *hash* criptográfica é resistente a colisão quando esta tornar inviável computacionalmente encontrar duas pré-imagens quaisquer que possuam o mesmo valor de *hash*. Neste caso, diferentemente da resistência a segunda pré-imagem, não é dado uma pré-imagem inicial para a qual precisa se achar uma segunda pré-imagem, mas é suficiente encontrar duas pré-imagens quaisquer tal que $H(M_1) = H(M_2)$.

Quando uma função *hash* é resistente a colisão, está é consequente resistente a segunda pré-imagem. Porém, nem sem sempre uma função resistente a segunda pré-imagem será resistente a colisão. Por isto, diz-se que uma função *hash* resistente a colisão é uma função de *hash* forte.

Se uma função *hash* não for resistente a colisão, então é possível para uma parte forjar a assinatura de outra parte. Por exemplo, se Alice deseja que Bob assine um documento dizendo que deve 100 reais a ela, caso Alice saiba que um documento contendo o valor de

1000 reais contém o mesmo valor de *hash* que o documento original, Alice pode fazer com que Bob seja responsável por uma dívida maior que a original, pois a assinatura valerá para ambos os documentos.

1.1.4 Uso das Propriedades de Funções *Hash*

Abaixo, temos uma tabela que mostra quais propriedades das funções *hash* são necessárias para alguma das aplicações de segurança de computadores:

Aplicação	Resistente a Pre-Imagem	Resistente a Segunda Pre-Imagem	Resistente a Colisão
Autenticação de Mensagens	X	X	X
Assinatura Digital	X	X	X
Infecção de Sistemas		X	
Arquivo de Senhas de Uma Via	X		

No caso da infecção de sistemas, não há problema em usar uma função de hash com fácil inversão, pois não é necessário embutir um valor secreto na geração do valor de *hash* de um arquivo. Já, num arquivo de *hash* de senhas, a inversão permitiria descobrir a senha a partir do valor de *hash*.

Se a função de *hash*, porém, permitir o descobrimento de uma segunda pré-imagem, seria possível infectar um arquivo de um sistema sem detecção, pois seu valor de *hash* não mudaria. Isto não seria um problema para um arquivo de *hash* de senhas, pois o perpetrante não possui a senha, que é a primeira pré-imagem e, portanto, não teria condições de descobrir a segunda pré-imagem.