Criptografia - SHA-256

Matheus Alves Alano Dias[1]

[1]Escola Politécnica
Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)
Av. Ipiranga, 6681 Partenon Porto Alegre - RS

matheus.dias96@edu.pucrs.br

1. Introdução

Neste relatório será descrito o processo de validação e autenticação de arquivos através de funções *hash*. Por exemplo, ao baixar um vídeo pela Internet, não é interessante esperar todo o arquivo ser baixado para então aplicar algum processo de validação. Então o processo que será realizado quebra o arquivo em blocos de 1024 bytes, aplica a função *hash* ao último bloco e concatena o valor ao penúltimo bloco. Esse processo é repetido até o primeiro bloco. Por fim, o *hash* do primeiro bloco é enviado ao usuário para que o mesmo possa validar os blocos enquanto o vídeo é baixado. Para implementar este trabalho, foi escolhida a linguagem de programação Python utilizando a biblioteca PyCrypto [pyc], que oferece as ferramentas necessárias para se trabalhar com a função *hash* SHA-256.

2. Carregamento do arquivo

Arquivos de vídeo no formato *.mp4* foram utilizados para testar e garantir o funcionamento do programa. Para abrir o arquivo em Python, foi utilizada a função *open()* em modo "rb", que significa *[r]eading as [b]inary*. A leitura do arquivo foi realizada utilizando a função *read()* passando por parâmetro os 1024 bytes, salvando posteriormente esse bloco em uma lista. Esse processo é repetido enquanto houver dados para ler do arquivo. O retorno dessa etapa é a lista de blocos de 1024 bytes, onde o último bloco pode ter tamanho inferior ao definido anteriormente.

3. Geração do hash

Após o carregamento do arquivo, inicia-se a etapa de geração do *hash*. Essa etapa requer uma lista de blocos na sua inicialização. Então essa lista de blocos é invertida para que o último bloco fique em primeiro lugar na lista.

Ao primeiro bloco dessa lista, é aplicada a função *hash*, e o resultado é salvo em uma variável chamada $last\_hash$. Ao iterar do segundo bloco em diante, é aplicada a função *hash* ao bloco com a variável $last\_hash$, salvando o resultado novamente na $last\_hash$.

Após esse processo de repetição por todos os blocos, o retorno dessa etapa é o valor atual da variável $last\_hash$.

4. Resultados

Ao concluir as duas etapas anteriores, é obtida a função *hash* do primeiro bloco do arquivo de vídeo. Para o arquivo de vídeo *video_03.mp4* descrito na especificação deste trabalho, o resultado obtido é:

$$ee24473e4a369a305c9c3d54629eff01f609b8e2f61ca9cf6f3084f13fe346d6 \quad (1)$$

Referências

Pycrypto - the python cryptography toolkit. https://www.dlitz.net/software/pycrypto/, Last accessed on 2019-09-28.