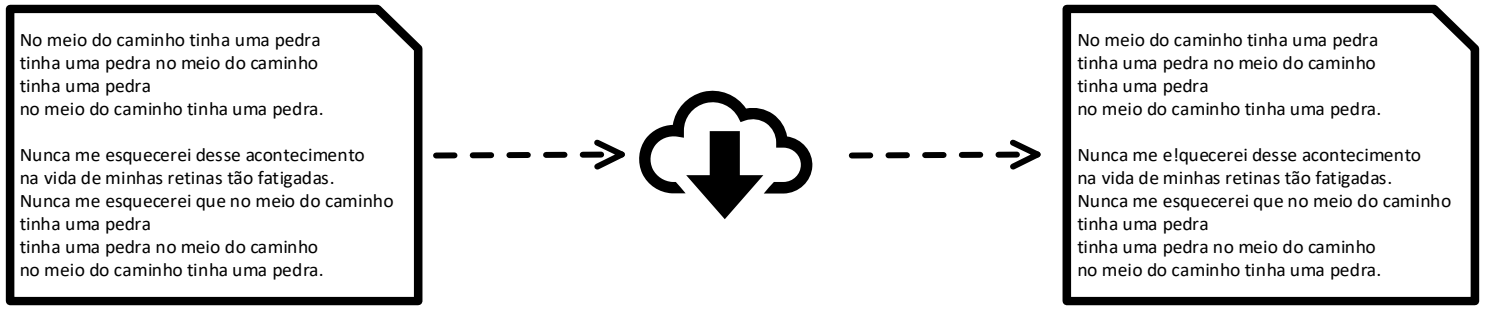
Funções para validar a integridade de dados

FURB
UNIVERSIDADE DE BLUMENAU

Motivação

O canal de comunicação pode ter ruídos ou pode ocorrer falha no armazenamento

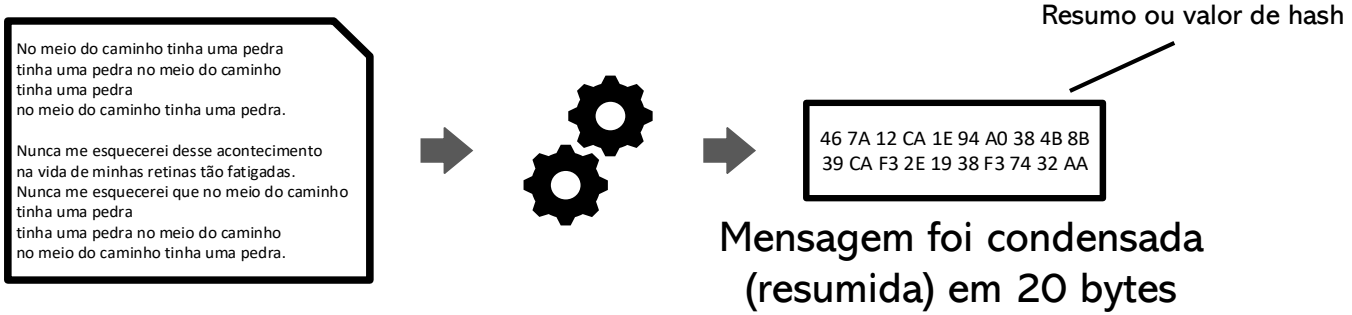
Como assegurar que o arquivo baixado é exatamente igual ao do servidor de origem?



Arquivo baixado difere ligeiramente do servidor

Resumo de mensagem

- Uma forma de identificar se os arquivos são íntegros é através da aplicação de um função de resumo de mensagem.
- O resumo de mensagem é um valor numérico computado a partir de um bloco de dados.



- O usuário baixa o arquivo e o resumo. Em seguida, gera o resumo de mensagem e confere com o resumo baixado

Resumo de mensagem

- Exemplo:

Daniel, eu vendi 4 máquinas para Satomi. Despache-as imediatamente.

39 5A 02 21 C1 BA 4C 64 54 02 AB 7C 1B CF CA 33 8C 27 8A 4B

Daniel, eu vendi 5 máquinas para Satomi. Despache-as imediatamente.

BB 76 D3 8F 62 3A 28 B2 A5 D8 D2 03 D6 5A CD D5 DD D6 97 61

- Os resumos “parecem aleatórios”
- Embora haja uma diferença de apenas um bit nas mensagens, os resumos são extremamente diferentes
- Não é possível reconstituir a mensagem original
  - Resumos são gerados por funções matemáticas unidirecionais.

Resumo de mensagem

- Um resumo de mensagem é um algoritmo que recebe qualquer comprimento de entrada e mescla a entrada para produzir uma saída pseudo-aleatória de largura fixa. (BURNET, 2002)

- Resumos de mensagem também são conhecidos como *funções hash de criptografia*, *message digest (MD)* ou simplesmente *hash.*

- Geralmente o valor do hash é divulgado em formato hexadecimal, como neste exemplo:

```
865c1d3b92e64a53f9f2e99c2a4c6af25a99ebc7c9a52c7ddf42dc5b31876
```

Resumo de mensagem

- Podem ser comparados à funções *checksum*, porém, ao contrário destas, não possui recurso para correção
  - As funções *checksum* foram projetadas para identificar falhas causadas por ruídos nos canais de transmissão. Não foram projetadas para detectar alterações causadas por tentativas intencionais e maliciosas
- Se fosse feito um ataque de força bruta, haveriam infinitas mensagens que levariam ao mesmo resumo.

Propriedades das funções de resumo de mensagem

- As funções de resumo de mensagem devem ter as seguintes propriedades:
  - São funções determinísticas – Uma mesma mensagem sempre produz o mesmo resultado
  - São rápidas de computar
  - Impossibilitam a geração de mensagem original a partir do valor de *hash*
  - Uma mudança pequena na mensagem deveria mudar drasticamente o seu valor de hash, de forma que o novo valor pareça não correlacionado com o valor de hash antigo.
  - Deve ser difícil encontrar duas mensagens diferentes m_1 e m_2 tal que $\text{hash}(m_1) = \text{hash}(m_2)$
    - A função deve ser resistente à colisão

Algoritmos de resumo de mensagem

- MD5 – Message digest algorithm 5
  - Produz um valor de hash de 128 bits (16 bytes)
  - Criado pela RSA Data Security
  - Possui pontos fracos e atualmente considerado impróprio

- SHA – Secure Hash Algorithm
  - Consiste numa família de algoritmos,
  - São similares ao MD5, porém sem as fraquezas do MD5

SHA – Secure Hash algorithm

- SHA-1
  - Criado em 1995
  - Produz um valor de hash de 160 bits (20 bytes)
- SHA-2
  - Publicado em 2001 como sendo o padrão federal para os Estados Unidos
  - SHA2 possui várias funções com resumos com valores de hash distintos:
    - SHA-224
    - SHA-256
    - SHA-384
    - SHA-512

SHA – Secure Hash algorithm

- SHA-3
  - Criado em 2015
  - Utiliza algoritmo não deriva do SHA-2
  - Possui variantes, chamadas de SHAKE128 e SHAKE256

Geração de resumo de mensagem em Java

MessageDigest
+ getInstance(algorithm : String) : MessageDigest + update(input : byte) : void + update(input : byte[], offset : int, len : int) : void + update(input : byte[]) : void + digest() : byte[] + digest(buf : byte[], offset : int, len : int) : int + digest(input : byte[]) : byte[] + isEqual(digesta : byte[], digestb : byte[]) : boolean + getDigestLength() : int

Algoritmos suportados
▪ MD2
▪ MD5
▪ SHA-1
▪ SHA-256
▪ SHA-384
▪ SHA-512

```
byte[] dados = "Mensagem de exemplo para geração de resumo".getBytes();

MessageDigest messageDigest = MessageDigest.getInstance("SHA-256");
byte[] digest = messageDigest.digest(dados);
```