PUC-Rio – Departamento de Informática
Cursos: Sistemas de Informação/
Ciência da Computação/
Engenharia da Computação
Disciplina: INF1416 – Segurança da Informação
Prof.: Anderson Oliveira da Silva

Trabalhos de Laboratório

Construir um programa Java que (i) use a JCA; (ii) não use interface gráfica; e (iii) seja executado em uma linha de comando com argumentos, da seguinte forma:

DigestCalculator <SP> Tipo_Digest <SP>Caminho_ArqListaDigest <SP>Caminho_da_Pasta_dos_Arquivos

onde,
Tipo_Digest – Tipo do digest a ser calculado (MD5 ou SHA1)
Caminho_ArqListaDigest - Informa a localização do arquivo que contém uma lista de digests de arquivos conhecidos.
Caminho_da_Pasta _dos_Arquivos - Informa a localização dos N arquivos que devem ser processados.
<SP> - Caractere espaço em branco.

O arquivo com a lista de digests utiliza o formato ASCII e é formado por zero ou mais linhas formatadas da seguinte maneira:

Nome_Arq<SP>Tipo_Digest<SP>Digest_Hex[<SP>TipoDigest<SP>Digest_Hex]<EOL>

onde,
Nome_Arq - Nome de um arquivo qualquer, sem informar o caminho.
TipoDigest - Indica o digest em seguida (MD5/SHA1/SHA256/SHA512).
Digest_Hex - Digest em hexadecimal referente ao tipo de digest especificado anteriormente.
<SP> - Caractere espaço em branco.
<*EOL*> - Caractere que marca o fim de linha (\n).
[ ] - O segundo digest pode ou não existir para um certo arquivo.

OBS: O arquivo que possuir mais de um digest registrado (MD5/SHA1/SHA256/SHA512) no arquivo de lista de digests deve ter apenas uma linha correspondente no arquivo com os dois digests, conforme a regra de formatação da linha.

O programa deve executar o seguinte procedimento:

1 - Calcular o digest solicitado do conteúdo de cada um dos N arquivos presentes na pasta fornecida;

2 - Comparar os digests calculados com os respectivos digests registrados para cada arquivo no arquivo ArqListaDigest, se existirem, e com os digests dos arquivos existentes na pasta;

3 - Imprimir na saída padrão uma lista com o seguinte formato:

Nome_Arq1<SP>Tipo_Digest<SP>Digest_Hex_Arq1<SP>(STATUS)
Nome_Arq2<SP>Tipo_Digest<SP>Digest_Hex_Arq2<SP>(STATUS)
. . . . . .
Nome_ArqN<SP>Tipo_Digest<SP>Digest_Hex_ArqN<SP>(STATUS)

onde:
<SP> - Caracter espaço em branco.
Nome_Arq1 .. Nome_ArqN - Correspondem aos N nomes dos arquivos encontrados na pasta fornecida para o cálculo dos digests (sem a informação do caminho da pasta).
Tipo_Digest - Tipo do digest calculado (MD5/SHA1/SHA256/SHA512)
Digest_Hex_ArqN – Digest formatado em hexadecimal calculado para o arquivo N.
STATUS - Corresponde a um dos status definidos abaixo:
OK = Status do arquivo cujo digest calculado é igual ao digest fornecido no arquivo ArqListaDigest e não colide com o digest de outro arquivo na pasta.
NOT OK = Status do arquivo cujo digest não é igual ao digest fornecido no arquivo ArqListaDigest e não colide com o digest de outro arquivo na pasta.
NOT FOUND = Status do arquivo cujo digest não foi encontrado no arquivo ArqListaDigest e não colide com o digest de outro arquivo na pasta.
COLISION = Status do arquivo cujo digest calculado colide com o digest de outro arquivo de nome diferente encontrado no arquivo ArqListaDigest ou com o digest de um dos arquivos presentes na pasta.

4 - Os digests calculados para os arquivos com status NOT FOUND devem ser acrescentados no final de uma linha existente para um nome de arquivo ou no final do arquivo de lista de digests para um nome de arquivo não existente, mantendo seu formato padrão. Os digests calculados para os arquivos com status COLISION não devem ser acrescentados no arquivo de lista de digests.

Observação 1: O nome do programa executável deve ser DigestCalculator.

Observação 2: O código fonte deve ser compilado com o Sun JDK 1.8.

Observação 3: Estude o método *update(byte[] input, int offset, int len)* da classe *MessageDigest* que atualiza o digest utilizando o array de bytes *input*, iniciando em *offset*.

Observação 4: Os digests devem ser calculados para o *conteúdo dos arquivos* presentes na pasta fornecida na linha de comando e NÃO para o *nome dos arquivos* que estão na pasta.

Observação 5: Se os argumentos da linha de comando forem omitidos ou insuficientes para a execução do programa, deve-se imprimir uma mensagem com a orientação de execução e, em seguida, o programa deve ser encerrado.

O programa fonte deve ser enviado como anexo via e-mail com o título INF1416-Gn: Trabalho 3, onde n identifica o número do grupo. Prazo de entrega: 29/4/2020 - 23:59h.