



Segurança Informática

Guia para Aula Laboratorial 5

Licenciatura em Engenharia Informática

Licenciatura em Informática Web

Licenciatura em Tecnologias e Sistemas da Informação

Sumário

Observação de propriedades importantes de funções de *Hash* através do cálculo de valores de resumo de vários ficheiros.

Pré-requisitos:

Algumas das tarefas propostas a seguir requerem o uso de *software* para efetuar cálculos e o acesso a um sistema que disponibilize a ferramenta OpenSSL, ou que permita a sua instalação. Sugere-se, assim, o uso de uma distribuição comum de Linux, onde todas estas condições estarão provavelmente preenchidas.

1 Utilização do OpenSSL para Obter os Valores do SHA1 e MD5 de um Ficheiro

Using OpenSSL to Obtain the SHA1 and MD5 Digests of a File

Nas últimas aulas, o OpenSSL tem sido usado para cifrar e decifrar alguns ficheiros, e até para gerar sequências de números aleatórios. Foi visto que o conjunto de funcionalidades desta ferramenta/*toolkit* é muito vasto, e os exercícios seguintes exploram apenas mais um subconjunto dessas funcionalidades.

Tarefa 1 *Task 1*

Faça o *download* do ficheiro cujo *Uniform Resource Locator* (URL) se inclui a seguir e obtenha os respetivos valores resumo para os algoritmos conhecidos como *Secure Hash Algorithm 1* (SHA1) e *Message Digest 5* (MD5):

podiaSerAFreqMasNaoE.pdf

Se precisar de ajuda a construir o comando, considere analisar a seguinte sopinha de letras, assinando os comandos/opções que lhe dariam jeito na construção do comando:

Computer Security

Guide for Laboratory Class 5

Degree in Computer Science and Engineering

Degree in Web Informatics

Degree in Information Technologies and Systems

Summary

Observation of important properties of hash functions via the calculation of digests of several files.

```
openssl -hex -in
ubi dgst -sdi
enc -sha1 -md5
```

```
openssl dgst -sha1 podiaSerAFreqMasNaoE.pdf
```

```
openssl dgst -md5 podiaSerAFreqMasNaoE.pdf
```

Compare os valores que obteve com os que são incluídos em baixo, e que foram os que o Professor efetivamente obteve quando fazia este guia laboratorial para o ficheiro indicado:

SHA1: bae9553f5138917dc765166f1ef7cd19784d5cc1

MD5: 812db75715dffae5119192cdace0980f

Q1.: Os valores são iguais ou diferentes?

☒ Iguais. ☐ Diferentes.

Q2.: Qual o significado da resposta anterior neste contexto?

- ☒ Que, neste caso, o ficheiro que chegou à minha máquina não sofreu erros durante a transmissão.
- ☐ Que, neste caso, o ficheiro que chegou à minha máquina sofreu erros durante a transmissão.
- ☒ Que, neste caso e confiando no que o Professor afirmou, o ficheiro que descarreguei é, de facto, o ficheiro que está no servidor, e para o qual calculou os valores de *hash*.

- ☐ Que, neste caso e confiando no que o Professor afirmou, o ficheiro que descarreguei não é, de facto, o ficheiro que está no servidor, e para o qual calculou os valores de *hash*.

Q3.: Quem é o criptógrafo que está por detrás do algoritmo MD5?

- ☐ Dan Boneh. ☒ Ron Rivest.
☐ Adi Shamir. ☐ Leonard Adleman.
☐ Bruce Schneier.

Q4.: Já conhecia este criptografo ou algum dos seus trabalhos?

- ☒ Sim, já conhecia pelo menos outra obra dele.

Qual? rc4

- ☐ Não, nunca ouvi falar dele...

Q5.: Quantos bits tem o resumo devolvido por cada uma das funções utilizadas?

MD5 : ☐ 64 ☒ 128 ☐ 192 ☐ 256
SHA1: ☐ 64 ☐ 128 ☒ 160 ☐ 256

Q6.: O que abrevia a letra S do acrónimo SHA1?

secure

Q7.: O que abrevia a letra D do acrónimo MD5?

digest

Tarefa 2 Task 2

Depois de ter preparado a frequência e a sua correção para este ano letivo, o docente calculou os respetivos valores resumos do ficheiro final, que transcreve para aqui:

SHA1: 476a2f642c7589adb4211b0dbe6401bc5c669b1c

MD5: 9b0e3ef62913177d0af1dbc622b041e0

Depois disso, o docente colocou o ficheiro com essa correção no *moodle*, para que pudesse ser descarregada pelos alunos. O ficheiro está disponível no link:

FrequenciaDestaCadeiraTodaTodaCorrigida.pdf

Depois de descarregar o ficheiro, calcule os valores de *hash* SHA1 e MD5. **Q8.: Os valores que obteve são iguais aos originais?**

- ☐ Sim, são. ☒ Não, são diferentes.

Q9.: Acha que o ficheiro pode ter sido modificado enquanto era transmitido ou que sofreu alterações?

- ☐ Sim, deve ter sido isso que aconteceu.
☐ Acho que, neste caso, não foi isso que aconteceu.

Creio que foi mesmo "*alguém*" que me quis enganar desta vez...

Q10.: Nas opções que se seguem, consegue identificar algumas utilidades para as funções de *hash*?

- ☒ Integridade dos dados.
☐ Confidencialidade.
☐ Fazer o pequeno-almoço.
☒ Identificação de burlas em ficheiros descarregados da Internet.

Tarefa 3 Task 3

Considere dar uma vista de olhos no seguinte URL:
<http://glua.ua.pt/pub/archlinux/iso/2018.10.01/sha1sums.txt>.

Q11.: O que é que este URL contém e qual a utilidade desse conteúdo?

Tarefa 4 Task 4

Crie um ficheiro chamado `original.txt` e coloque bastante texto lá dentro. Calcule e guarde os valores de *hash* desse ficheiro com vários algoritmos.

SHA1: b63411cca2dc1af4f985f040b483cccf76b01060

MD5: 8733bbefbc0fb7880de115b4a4396531

Para referência futura, escreva aqui os comandos que utilizou para calcular os valores acima indicados:

`openssl`

Abra novamente o ficheiro, e **altere uma só letra do texto**, guardando-o com o nome `alterado.txt`. Se quiser ser mesmo *picuinhas*, use o programa fornecido na aula anterior para alterar um único bit do ficheiro. Calcule e compare os valores de *hash* deste ficheiro com os do original.

SHA1 (original) b63411cca2dc1af4f985f040b483cccf76b01060

SHA1 (alterado) cefb2d81c38050f663f54a80fddf011c945de62b

MD5 (original) 8733bbefbc0fb7880de115b4a4396531
MD5 (alterado) 425678f615caedd1317b59fd30e85144

Q12.: O que é que aconteceu após a alteração de um único byte?

- ☐ Os valores de *hash* ficaram iguais.
- ☐ Os valores de *hash* ficaram praticamente iguais.
- ☐ Os valores de *hash* ficaram um pouco diferentes.
- ☒ Os valores de *hash* ficaram diferentes.
- ☐ Os valores de *hash* ficaram totalmente diferentes.

10110110001101000001000	11001110111110110010101
111001100101000101101110	10000001110000111000000
000011010111101001111100	00101000011110110011000
1100001011110000010000	11111101010100101010000
00101101001000001111001	0001111101101111100000
100110011110111011010110	00100011100100101000101
0000001000001100000	11011110011000101011

Tarefa 5 Task 5

Converta os valores obtidos anteriormente em hexadecimal para binário e volte a compará-los.

Reveja a sua resposta à questão anterior, e atente na seguinte. **Q13.: Quantos bits ficaram diferentes após a alteração de um único byte?**

- ☐ Ficaram todos iguais.
- ☐ Ficaram todos diferentes.
- ☐ Mudaram aproximadamente 1/4 dos bits.
- ☒ Mudaram aproximadamente 1/2 dos bits.

Q14.: O que é que aconteceria se se mudassem vários bytes em vez de um só?

- ☐ Ahh. Assim os valores de *hash* já mudavam por inteiro.
- ☐ Obtínhamos um efeito de cancelamento se o número de bytes alterado fosse par, e os valores de *hash* ficavam iguais.
- ☐ Obtínhamos valores de *hash* em que já mudavam mais do que só metade dos bits.
- ☒ Obtínhamos exatamente o mesmo comportamento que observamos quando mudamos 1 só byte.

Q15.: Parece-lhe haver alguma relação entre o byte alterado no ficheiro e os bits que foram alterados nos respetivos valores de *hash*?

- ☒ Não. As alterações produzidas parecem ser aparentemente imprevisíveis.
- ☐ Sim. Parece que ao alterar o 3 byte do texto, também altero sempre o 3 bit do valor de *hash*.
- ☐ Sim, consigo notar uma relação entre a alteração de qualquer byte e vários bits do valor de *hash*, mas é muito muito complexa, e só a minha cabeça é que consegue perceber essa relação.

Q16.: Acha que conseguia alterar o ficheiro original de tal forma que ia obter os mesmos valores de *hash* que o original?

- ☐ Sim, porque não?

☒ Não estou a ver como iria fazer isso para já, mas creio que iria conseguir. A minha ideia era a seguinte

Q17.: A que propriedade das funções de *hash* criptográficas se refere a questão anterior?

- ☐ Resistência a colisões.
- ☐ Resistência à descoberta de um texto original.
- ☒ Resistência à descoberta de um segundo texto original.

Q18.: Acha que conseguia ao menos arranjar dois ficheiros diferentes com o mesmo valor do SHA1 em tempo útil?

- ☐ Se sim, faça-o.
- ☒ Se não, procure explicar.

O SHA1 é uma função de *hash* criptográfica, logo seria impossível, na prática, encontrar 2 ficheiros diferentes com o mesmo valor de *hash*.

Q19.: A que propriedade das funções de *hash* criptográficas se refere a questão anterior?

- ☒ Resistência a colisões.
- ☐ Resistência à descoberta de um texto original.
- ☐ Resistência à descoberta de um segundo texto original.

Tarefa 6 Task 6

Por vezes, para não deixar as palavras-chave (*passwords*¹) numa forma legível guardadas num ficheiro, guarda-se a representação obtida através da transformação dessas palavras-passe por via de funções de *hash* (ou funções de derivação de chaves). A forma mais simples será guardar o valor de *hash* dessa palavra-chave. Neste caso, de cada vez que se quer verificar se a palavra-chave introduzida está correta, calcula-se primeiro o seu valor de *hash* e compara-se com o que foi guardado.

Considere que determinado site implementa a versão mais simples do mecanismo mencionado antes, i.e., são guardados os MD5 de todas *passwords* num ficheiro, em vez das *passwords*. Considere que um larápio conseguiu aceder ao sistema e roubar o ficheiro das *passwords*. Entre elas, estava o seguinte valor MD5, que corresponde ao utilizador chico-fininho:

c6cc8094c2dc07b700ffcc36d64e2138

¹Nota: *password* não é o mesmo que chave de cifra.

Q20.: Consegue descobrir a palavra-passe que o utilizador *chico-fininho* usa?

- ☒ Sim, é _____.
- ☐ Não, neste caso é impossível.

Sugestão: caso esteja com dificuldades em responder à questão, considere visitar o seguinte URL <http://md5.gromweb.com/> para responder a esta questão.

Q21.: Dado a resposta que deu antes, acha que o método *simples* enunciado em cima é seguro?

- ☐ Sim, é seguro.
- ☒ Não, porque não protege contra ataques que usam *rainbow tables* ou tabelas de *hash* pré-computadas. Ora bolas! :S

Procure métodos mais seguros que o que foi enunciado antes (e.g., em https://passlib.readthedocs.io/en/stable/lib/passlib.hash.md5_crypt.html).