Assignment 1

CMPD333 Fundamental of Cyber Security

Diploma in Computer Science

Prepared for: Nor Nashrah Azmi

Prepared by:

No	ID	Name
1	DC98883	SITI FATIMAH ZAHRA BINTI ABD RAHMAN
2	DC98871	AININ SOFIYA BINTI MOHD ZARAK ZURKANAIN

Last Updated:
12/4/2024

Part 1

Original.txt File :

A hash function is any function that can be used to map data of arbitrary size to data of fixed size. The values returned by a hash function are called hash values, hash codes, hash sums, or simply hashes. One use is a data structure called a hash table, widely used in computer software for rapid data lookup. Hash functions accelerate table or database lookup by detecting duplicated records in a large file. An example is finding similar stretches in DNA sequences. They are also useful in cryptography. A cryptographic hash function allows one to easily verify that some input data maps to a given hash value, but if the input data is unknown, it is deliberately difficult to reconstruct it (or equivalent alternatives) by knowing the stored hash value. This is used for assuring integrity of transmitted data, and is the building block for HMACs, which provide message authentication.
Hash functions are related to (and often confused with) checksums, check digits, fingerprints, randomization functions, error-correcting codes, and ciphers. Although these concepts overlap to some extent, each has its own uses and requirements and is designed and optimized differently. The Hash Keeper database maintained by the American National Drug Intelligence Center, for instance, is more aptly described as a catalogue of file fingerprints than of hash values.

Hash Output :

MD5 : b4019c54bdaec977b3f577ff73335abc
SHA1 : 9910b6082af054a55dac7a5d0e0f09b310180905
SHA256 : 2dcbdbb30324aff15e8061abee730ae70fdaf97b9c4d8606ac3ee86400a0d8e6

Modified.txt File :

A hash function is any function that can be used to map data of arbitrary size to data of fixed size. The values returned by a hash function are called hash values, hash codes, hash sums, or simply hashes. One use is a data structure called a hash table, widely used in computer software for rapid data lookup. Hash functions accelerate table or database lookup by detecting duplicated records in a large file. An example is finding similar stretches in DNA sequences. They are also useful in cryptography. A cryptographic hash function allows one to easily verify that some input data maps to a given hash value, but if the input data is unknown, it is deliberately difficult to reconstruct it (or equivalent alternatives) by knowing the stored hash value. This is used for assuring integrity of transmitted data, and is the building block for HMACs, which provide message authentication.
Hash functions are related to checksums, check digits, fingerprints, randomization functions, error-correcting codes, and ciphers. Although these concepts overlap to some extent, each has its own uses and requirements and is designed and optimized differently. The Hash Keeper database maintained by the American National Drug Intelligence Center, for instance, is more aptly described as a catalogue of file fingerprints than of hash values.

Hash Output :

MD5 : ab5130ff1895515e77b750dc635c5bb0
SHA1 : 68133eab565a6fb8fd0435cbfd89586e5d0a0b06
SHA256 : 3fa09eb2ff6454251e22fb7e073603be1fe2bf066acc2f4c6bd99b3cd9faa70c

Conclusion : The different in hash codes indicate that even a small modification in the file content can result in significant changes in the hash values. This phenomenon is a fundamental property of cryptographic hash functions.

Part 2

Original Text :

1. Siti Fatimah Zahra Binti Abd Rahman :

MD5 : c30081d2539ea1d0b69570291c9247f3

SHA1 : 25722d78dbd802be5ffcf47e8b8c0e46bd993788

SHA256 : 33282a094ecdb7e9954053d01885d0647d6e60af216697d24d17f1dbb185abd2

2. Ainin Sofiya Binti Mohd Zarak Zurkanain :

MD5 : b7943bd676999a0ae3f4795060b3b089

SHA1 : 7e962d4c799bdc106b1fc524f389ecfffba0530c

SHA256 : a96ccba45efa0a900e16a0645b270c24498ce5e2b455e0d65de779a0ad4f80db

Part 3

1. Siti Fatimah Zahra Binti Abd Rahman

Algorithm : DES Mode : ECB Key : TOP

Encrypted text:

```
00000000  e9 8c 39 a4 3d c9 85 c2 ec 6a 4a 62 2e 2e c7 b0  | é . 9 ¤ = É ▯ Â ì j J b . . Ç °
00000010  f4 1d 88 e5 a4 54 a5 d6 3b 24 2a 18 95 82 2e fa  | ô . ▯ å ¤ T ¥ Ö ; $ * . ▯ . . ú
00000020  fb c5 83 7e 00 61 20 56                          | û Å . ~ . a   V
```

Algorithm : DES Mode : ECB Key : SECRET

Encrypted text:

```
00000000  4c 8e f0 e6 2d a8 72 bf 80 30 d6 5e 60 7c 80 22  | L . ð æ - ¨ r ¿ . 0 Ö ^ ` | . "
00000010  46 2b 1e a3 61 5b 09 16 16 36 05 b2 ce 93 d2 38  | F + . £ a [ . . . 6 . ² Î . Ò 8
00000020  37 c8 6b f4 07 42 7e 2e                          | 7 È k ô . B ~ .
```

Algorithm : 3DES Mode : ECB Key : TOP

Encrypted text:

```
00000000  e9 8c 39 a4 3d c9 85 c2 ec 6a 4a 62 2e 2e c7 b0  | é . 9 ¤ = É ▯ Â ì j J b . . Ç °
00000010  f4 1d 88 e5 a4 54 a5 d6 3b 24 2a 18 95 82 2e fa  | ô . ▯ å ¤ T ¥ Ö ; $ * . ▯ . . ú
00000020  fb c5 83 7e 00 61 20 56                          | û Å . ~ . a   V
```

Algorithm : 3DES Mode : ECB Key : SECRET

Encrypted text:

```
00000000  4c 8e f0 e6 2d a8 72 bf 80 30 d6 5e 60 7c 80 22  | L . ð æ - ¨ r ¿ . 0 Ö ^ ` | . "
00000010  46 2b 1e a3 61 5b 09 16 16 36 05 b2 ce 93 d2 38  | F + . £ a [ . . . 6 . ² Î . Ò 8
00000020  37 c8 6b f4 07 42 7e 2e                          | 7 È k ô . B ~ .
```

Algorithm : AES Mode : ECB Key : TOP

Encrypted text:

```
00000000  ab df b3 50 0a 21 1a 68 d4 2e 85 39 0f 85 37 af  | « ß ³ P . ! . h Ô . ▯ 9 . ▯ 7 ¯
00000010  1d 59 91 81 3d ad 1e 08 63 c2 dc 12 3d c3 fc 5d  | . Y ▯ . = - . . c Â Ü . = Ã ü ]
00000020  81 e8 11 f2 22 be 62 21 47 59 ab a6 31 68 bc 3a  | . è . ò " ¾ b ! G Y « ¦ 1 h ¼ :
```

Algorithm : AES Mode : ECB Key : SECRET

Encrypted text:

```
00000000  da a8 ca 77 e9 17 62 66 ab d4 51 fa 0d d6 fa b3  | Ú ¨ Ê w é . b f « Ô Q ú . Ö ú ³
00000010  e7 27 0b 84 da 45 9f 6b 54 8d fd 24 6e 49 45 85  | ç ' . . Ú E . k T ▯ ý $ n I E ▯
00000020  0c 4d 43 11 79 fe 5b 0b 10 13 27 7e 49 32 05 e9  | . M C . y þ [ . . . ' ~ I 2 . é
```

2. Ainin Sofiya Binti Mohd Zarak Zurkanain

Algorithm : DES Mode : ECB Key : TOP

Encrypted text:

```
00000000  3f 66 50 d4 42 04 3a 56 fe dc 3d ca ef a5 49 89
00000010  3c 41 eb 47 04 bb 0f 40 2a 7d 50 be 1e 4d 33 08
00000020  8e 70 f4 d0 82 77 09 52
```

Algorithm : DES Mode : ECB Key : SECRET

Encrypted text:

```
00000000  d1 80 b1 01 77 3d 04 84 6f a3 c2 86 a8 60 97 16
00000010  db f2 4c 4b 39 f1 7a 40 b4 f6 74 4d cd bc 69 c5
00000020  a1 b0 be 5f 4a 13 af b7
```

Algorithm : 3DES Mode : ECB Key : TOP

Encrypted text:

```
00000000  3f 66 50 d4 42 04 3a 56 fe dc 3d ca ef a5 49 89
00000010  3c 41 eb 47 04 bb 0f 40 2a 7d 50 be 1e 4d 33 08
00000020  8e 70 f4 d0 82 77 09 52
```

Algorithm : 3DES Mode : ECB Key : SECRET

Encrypted text:

```
00000000  d1 80 b1 01 77 3d 04 84 6f a3 c2 86 a8 60 97 16
00000010  db f2 4c 4b 39 f1 7a 40 b4 f6 74 4d cd bc 69 c5
00000020  a1 b0 be 5f 4a 13 af b7
```

Algorithm : AES Mode : ECB Key : TOP

Encrypted text:

```
00000000  0e 43 48 17 af 15 bc 8e 0f c0 52 da 94 dc 8d 03
00000010  3c 9c e0 4b 0b 07 b3 84 fd 69 a2 e6 8d c2 c9 d3
00000020  53 a7 f0 33 24 0a 26 2a d8 fc 91 3c 3e 22 3a c5
```

Algorithm : AES Mode : ECB Key : SECRET

Encrypted text:

```
00000000  24 e7 7d 7c a5 66 6c fc 57 7b fa 46 6d b0 f3 55
00000010  b4 01 04 55 f1 6a 54 c6 32 f9 06 75 1b 6b 77 bf
00000020  47 a1 8f 04 e7 b0 d4 b2 58 b5 57 ba e4 12 5a 15
```

Conclusion : The different in cipher text demonstrate the impact of the encryption key on the resulting ciphertext. Even though the encryption algorithm and mode remain the same, changing the key produces entirely different ciphertext.