What is Encryption/Decryption?

In computer terms, encryption is a process of scrambling data from sensible/readable format into non sensible/readable format with the help of key. Decryption is the other way round, from nonsensible/readable format into sensible/readable format with the help of key. If any parties possess the key, they can read the message else they can't.

The example is, assume you have an important physical valuable asset, you put that asset into a lockable box and locks it, this called encryption. If you unlock the lock on the box and take the asset out, this called decryption.

What is Single Key Encryption (Symmetric Encryption)?

Let's imagine you have a box, a lock and a key. The lock was built-in to the box which it uses only a single key to lock/unlock the box. For the sake of argument, the box, the lock and the key is indestructible.

When you want to lock or unlock the box, you use only a single key to do it.

You can imagine the key was your password as it's the easiest way to understand. You use your password to lock or unlock the box.

If locking and unlocking the box involves with only a single key such as password, this was called as single key encryption or in official terms it was called symmetric encryption.

We are obviously not dealing with boxes in real world. This is just the concept for you to understand.

Single key encryption or symmetric encryption is generally used in internet or in computer world. People prefer to use symmetric encryption because the speed in performing encryption or decryption is extremely fast.

However, there's a problem in using symmetric encryption. Symmetric encryption uses only a single key such as password to perform both encryption and decryption. Users have to consider how to deliver the password or the key to the recipient safely. If you use (exclude end to end encryption) email such as Google mail or Microsoft email/phone calling or video calling or voice calling, these options are not recommended because the potential key or password could be known by third party.

The security of symmetric encryption relies on the users to keep the key or password safe, confidential and secure. Any attacker can't break the "box or the lock built-in to the box"- symmetric encryption algorithm, the only possible attack they could do is to steal or to guess your key/password.

There're more details but I will leave it for you to discover.

What is bits?

Computers are generally operating in binary number system (0,1) whereas human uses decimal or base 10 number (0,1,2,3,4,5,6,7,8,9) system. Bits are one of the computer terms.

Let's look at an example. When we want to pay 2 dollar to a convenient store staff, we take out our wallet and count 1 dollar.. 2 dollar. Once we think that there's enough money we proceed to give the staff 2 dollar as a payment.

In human world we can count up from 0,1,2,3,4,5,6,7,8,9,10,11……., this is how we count numbers and how we deal with numbers.

In computer world, it was represented differently. What you need to look at was the computer representation of numbers after the "?-" symbols.
0?-00, 1?-01, 2?-10, 3?-11
4?-100, 5?-101, 6?-110, 7?-111
8?-1000, 9?-1001, 10?-1010, 11?-1011, 12?-1100

Let us use the above reference as an example. **0/1/2/3** has 2 bits, **4/5/6/7** has 3 bits, **8/9/10/11/12** has 4 bits. Bit(s) is essentially how many reserved space that are responsible for storing the different combination of 0 or 1. People can say 1 bit, it just mean there's 1 reserve space that can either store 0 or 1. When people say 2 bits, it means there're 2 reserve spaces that can either store all 0 or all 1 or a combination of 0 and 1.

Symmetric encryption type 1-Block Cipher

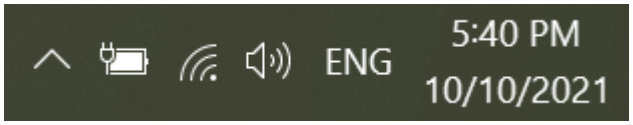
Let's assume we want to encrypt a message of "The first thing ", if you count the number of characters (include spaces) in the message, you will see that it's exactly 16 characters. Block cipher refers to you take exactly 16 characters and perform encryption on the 16 characters and produces encrypted text which looks similar to "8Rs6GxTgG1RZhrJe/li6Pg==" (Base 64) or "F11B3A1B14E01B545986B25EFE58BA3E" (Hexadecimal number).

For decryption you take the encrypted text which could be very long and block cipher will also use the same way that it uses on performing encryption to perform decryption.

For more information, you can refer to (https://www.devglan.com/online-tools/aes-encryption-decryption).

Symmetric encryption type 2-Stream Cipher

When we buy a computer, it usually comes along with a keyboard. Before you switch the input language from English to other language, all the possible characters, letters and symbols that you can type and it shows on the screen was "ASCII" text.



The full name of "ASCII" is American Standard Code for Information Interchange.

"ASCII" letters/symbols/characters generally use 8 bits.

"Unicode" words or symbols or emoji generally uses 32 bits.

We won't go over to too much detail, let's stick to ASCII.

Assuming the text we want to encrypt was "The first thing ".

The letter "T" uses 8 bits.
The letter "h" uses 8 bits.
The space uses 8 bits.

Once I convert the letter "T" into computer format, it's represented as "01010100".

We will perform XOR operation on the letter "T" in its computer format.

Let's suppose there're 2 computer formatted text.
"01010100"
"01111010"

XOR operations defined as below.
0+0 = 0
1+1 = 0
0+1 = 1

If we perform XOR operation on those 2 computer formatted text. We should get a final output that's shown below.
"00101110".

This is in general how an XOR operation works.

Let's say we use password to encrypt the message. Assume that our password was "Password12345678", we convert it into computer formatted text.

The message in computer formatted text was shown below.
01010100011010000110010100100000011001100110100101110010011100110111
01000010000001110100011010000110100101101110011001110010000 0

Sample password in computer formatted text was shown below.
01010000011000010111001101110011011101110110111011100100110010000110
00010011001000110011001101000011010100110110001101110011100 0

Stream cipher encrypts the message by performing XOR operation on the computer formatted password/key with the message. If someone uses stream cipher to encrypt a message, we can just do “encryption” again to get back the original message by performing XOR operation on the computer formatted password/key with the encrypted message.

Here’s a sample run (Encrypted Message).
00000100000010010001011001010011000100010000011000000000000101110100
01010001001001000111010111000101110001011000010100000000011000

Here’s a sample run (Decrypted Message)
01010100011010000110010100100000011001100110100101110010011100110111
01000010000001110100011010000110100101101110011001110010000 0

You may want to try on your own to see how it works. Here’s the links.
(https://toolslick.com/math/bitwise/xor-calculator)
(https://www.binaryhexconverter.com/ascii-text-to-binary-converter)

The second link was responsible to convert ASCII text to computer formatted text. You can put in any message you want to encrypt then get its computer formatted form.
You can also put in any password you want to use then get its computer formatted form as well.

Once you get 2 different computer formatted text, you can perform encryption/decryption by XOR with each other.

In modern industry, stream cipher is too risky to use on its own. Hence, when we use stream cipher, we generally use it in conjunction with MAC or message authentication code. We use cryptography hashing in a special way to produce MAC. However, it’s up to you whether you want to further study on this specific matter.