cryptography.html

tableOfContents.css

# Cryptography {

## [Encryption & Decryption]

Using XOR

By : < AmirHossein Heidari >

MaxEdison

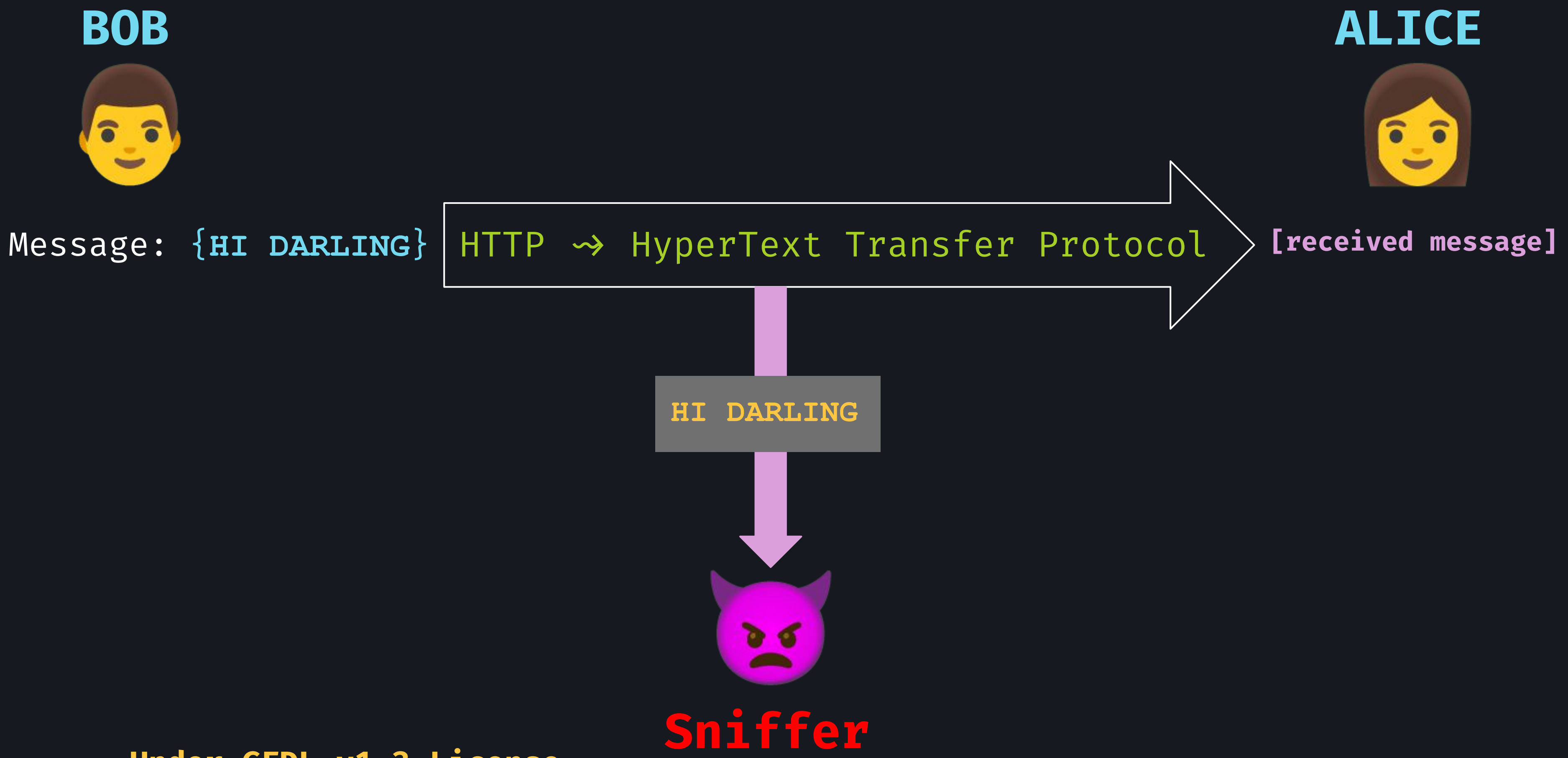
MaxEdison

}

# Contents Of ‘This Presentation’;

Here’s what you’ll see in this presentation:

- **The Scenario**
- **What is CRYPTOGRAPHY and why we must use it?**
- **Caesar Cipher** - Elementary Cryptography Algorithm
- **Computer Science** and **Mathematics** correlation
- What is **XOR** operator and how we use it in **cryptography**?



BOB

ALICE

Message: {HI DARLING}

HTTP ⇝ HyperText Transfer Protocol

[received message]

HI DARLING

Sniffer

cryptography.html

commonMistakes.css

# Cryptography:

is a very broad science.

is child of Mathematics.

### Abstraction

Create a puzzle based on Mathematics principles and hide content in the puzzle.

NO ONE in the world is able to solve the puzzle, cause they don’t have information (the KEY).

Just the person who has the information (the key) is able to solve it (Decrypt the cipher).

Mathematics provides the foundation for cryptography. The secure communication and protection of sensitive information rely on mathematical concepts and principles such as ***modular arithmetic***, ***prime numbers***, ***number theory***, ***linear algebra***, ***probability theory,*** and ***information theory.***

# ‘Common Mistakes’ {

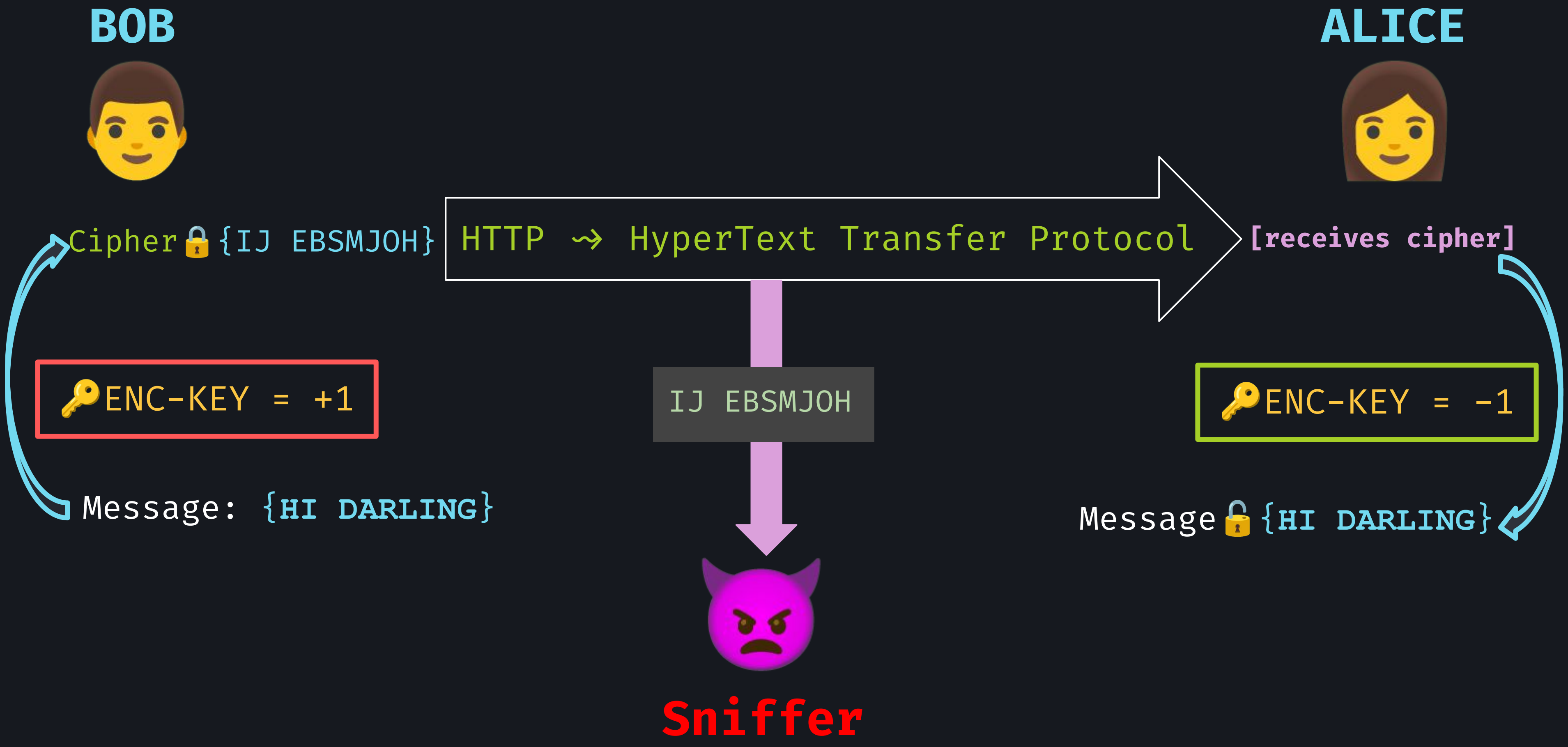
## 01 Hashing is NOT Encrypting

< Don’t make mistake! Hash is different from cryptography >

## 02 Encoding is NOT Encrypting

< Encodings like ASCII or PNG are not related to encryption at all !!! >

}



BOB

ALICE

Cipher🔒{IJ EBSMJOH}

HTTP ⇝ HyperText Transfer Protocol

[receives cipher]

🔑ENC-KEY = +1

IJ EBSMJOH

🔑ENC-KEY = -1

Message: {HI DARLING}

Message🔓{HI DARLING}

Sniffer

csAndMath.html result.css

```
< /ComputerScience > {

    < Computer has a `Discrete Entity`.
    Better to say it is finally  0 and 1s. >

}

< /Mathematics > {
    < The concept corresponding to 0 and 1
    in the mathematical world is base two
    (Binary). >
}
```

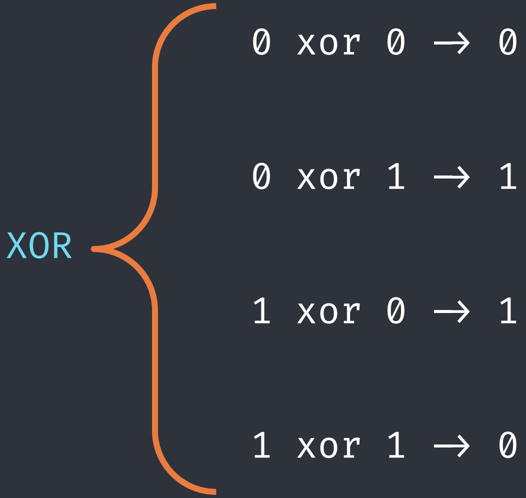
So, if we want to Encrypt any content, we should manipulate its Binary Code!

Now Let's check a very Simple Encryption (also Decryption) Algorithm, which is practical in Real World !

XOR.html    encryptWithXOR.css

# XOR (Inequality Detector); {

‘Explaining how XOR operator works’



XOR

0 xor 0 → 0

0 xor 1 → 1

1 xor 0 → 1

1 xor 1 → 0

}

XOR.html

encryptWithXOR.css

# How to Encrypt with ‘XOR’? {

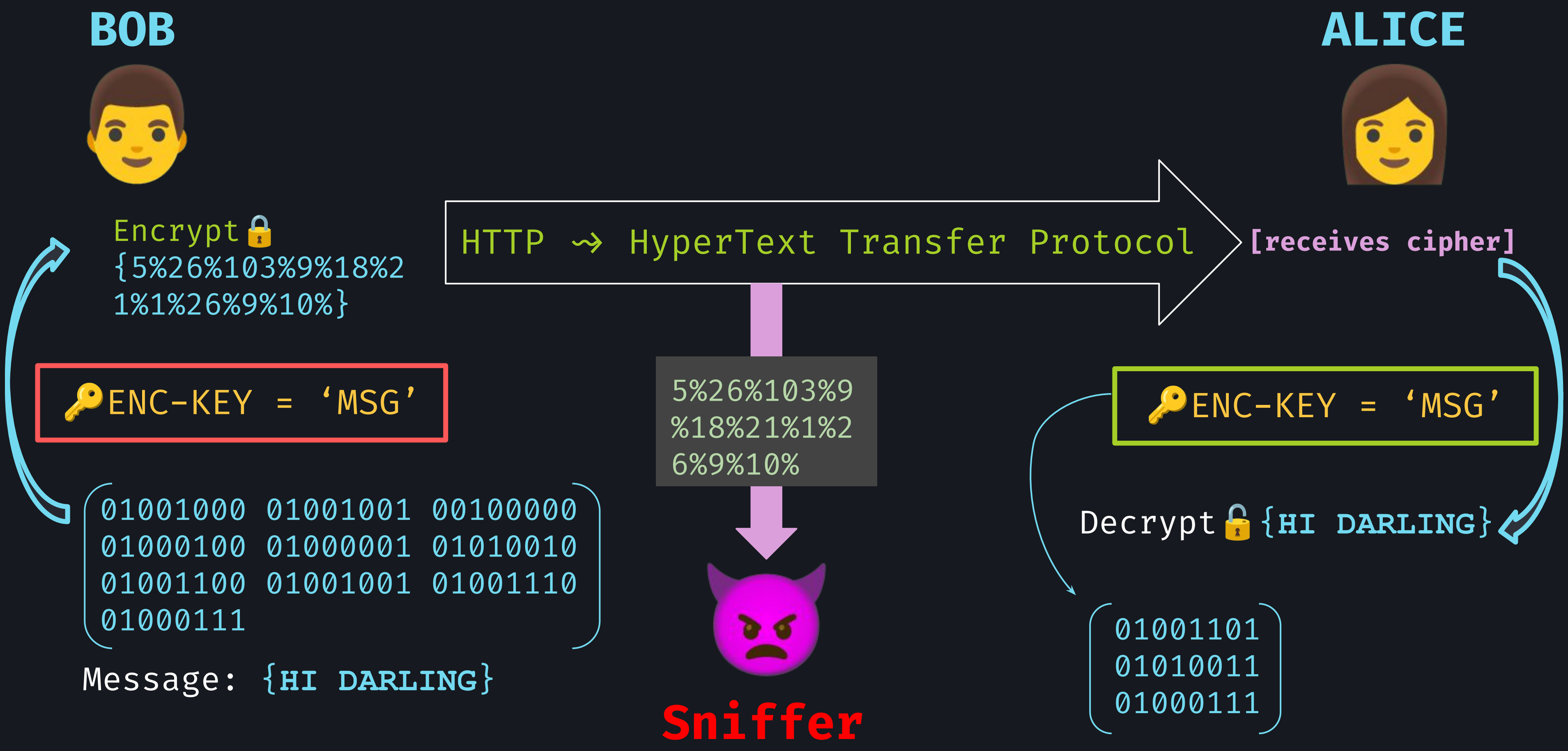
Step 00 First, we should declare a KEY. I choose ‘MSG’ for this case. -Two sides should have this KEY-

Step 01 Convert both KEY and message content to their BINARY value.

Step 02 XOR BIN message with BIN KEY, Byte by Byte. XOR first Byte of message with first Byte of KEY, then second Bytes and so on...

Step 03 Now, you can Encode your Encrypted data to ASCII or anything else and send it through protocols.

}



**BOB**

**ALICE**

Encrypt🔒
{5%26%103%9%18%2
1%1%26%9%10%}

HTTP ⇝ HyperText Transfer Protocol

**[receives cipher]**

🔑ENC-KEY = ‘MSG’

```
01001000 01001001 00100000
01000100 01000001 01010010
01001100 01001001 01001110
01000111
```

Message: {**HI DARLING**}

5%26%103%9
%18%21%1%2
6%9%10%

**Sniffer**

🔑ENC-KEY = ‘MSG’

Decrypt🔓{**HI DARLING**}

```
01001101
01010011
01000111
```

The Simple Encrypt/Decrypt Using XOR (SEDUX) Program is available on my GitHub profile:

github.com/MaxEdison/SEDUX

## Created with ❤️ - AmirHossein Heidari

Thanks to @TadavomnisT

2024

Powered By: slidesgo