(Applied) Cryptography

Tutorial #3

Bernardo Portela (bernardo.portela@fc.up.pt) Rogério Reis (rvreis@fc.up.pt)

MSI/MCC/MERSI – 2025/2026

Create a file that is at least 1000 bytes long. A text file is the most straightforward way to do this, but you can use any file of such size.

1 - Use Python to encrypt this file in AES-CBC mode and decrypt it. Check for success

(ref: https://cryptography.io/en/latest/hazmat/primitives/symmetric-encryption/).

2 - Repeat this process with OpenSSL

(ref: https://www.openssl.org/docs/man1.1.1/man1/enc.html).

3 - After encryption, edit the file to change the value of (but not delete!) one byte and decrypt again.

3.1 - What happened? How much of the file was corrupted by this change in the ciphertext?

3.2 - Could you recover a file encrypted with CBC if the IV and the first ciphertext block were corrupted (i.e. changed to other values)?

3.3 - Could you recover it if during a satellite transmission the first bit of the ciphertext is not delivered? This means that the message that was supposed to be n bits long is instead $n - 1$ bits long.

3.4 - Suppose you encrypted a very large message using AES-CBC. You now realize the first byte of the message was incorrect! (you wrote “hello” instead of “Hello”!)

Can you modify the existing ciphertext to reflect this change, or do you have to re-encrypt the whole message? Justify.

4 - Repeat the exercise with CTR mode. What are the differences?