Lab 2

- **cilj kriptografije** → povjerljivost i autentičnost (integritet) poruke koja se šalje kom.kanalom
- **povjerljivost poruke** → čitljiva samo onom kome je namijenjena, a ostalima neupotrebljiva → izvorni tekst se pretvara u šifrirani tekst i šalje kom. kanalom
- **integritet poruke** → osigurava da podaci (sadržaj poruke) nisu izmijenjeni od strane neke treće osobe → ostvaruje se MAC-ovima
- integritet poruke < = > autentifikacija izvora
- **MAC** ("Message Authentication Code") → mehanizam kojim se provjerava integritet poruke temeljen na tajnom ključu
  - niz bitova koji se dodaju na originalnu poruku u cilju očuvanja integriteta poruke i autentifikacije pošiljatelja

- način korištenja ključa
  - simetrični i asimetrični algoritmi kriptiranja
- **simetrični** → **isti ključ** za enkripciju i dekripciju podataka
  - AES("Advanced Encryption Standard") → ključevi duljine 128,192 i 256 bita
  - Osnovni el. za simetricnu enkripciju:

  ▼ **plaintext(P)** → originalna poruka koja se enkriptira

  ▼ **encryption algorithm(E)** → primjenjuje radnje na plaintext-u

  ▼ **secret key (K)**

  ▼ **decryption algorithm(D)** → pomoću ciphertexta i sigurnosnog ključa dolazi do originalne poruke
  - Napadi na simetričnu enkripciju:
    1. Kriptoanaliza
    2. **Brute-force napad**

- U vježbi se za dekriptiranje personalizirane enkriptirane poruke koristi brute-force napad
- Brute-force napad → univerzalan,izravan napad
  - Isprobava se svaki mogući ključ te se uspoređuje je li enkriptirana poruka ,dekriptirana pomoću odgovarajućeg ključa, jednaka početnom tekstu
  - **Izvođenje vježbe:**
    1. pozicioniranje u direktorij
    2. **python -m venv_(ime)** → stvaranje direktorija za izvršne datoteke koje će se upotrijebiti prilikom izrade projekta u pythonu
    3. **pip install cryptography** → u pozicioniranom direkoriju se stvara biblioteka naziva cryptography
    4. uključivanje biblioteke Fernet
    5. **generate_key()** → generira se sigurnosni ključ
    6. **f.encrypt** → enkriptira se tekst
    7. **brute_force()** → fja. koja se definira u pythonu,tamo se otvara datoteka sa zadatkom te se čita naredbama open i read
       a. funkc. **while** se omogućava dekripcija na način da se isprobavaju svi mogući sigurnosni ključevi sve dok se ne nađe onaj kojim se dolazi do plaintext-a

- **hash** funkcija → sažimanje i identificiranje podataka
  - podatke promjenjive veličine pretvara u podatke fiksne veličine
  - **primjena: očuvanje integriteta poruka**