

## 2. laboratorijska vježba : Symmetric key cryptography

Studentica: Emilija Sarić

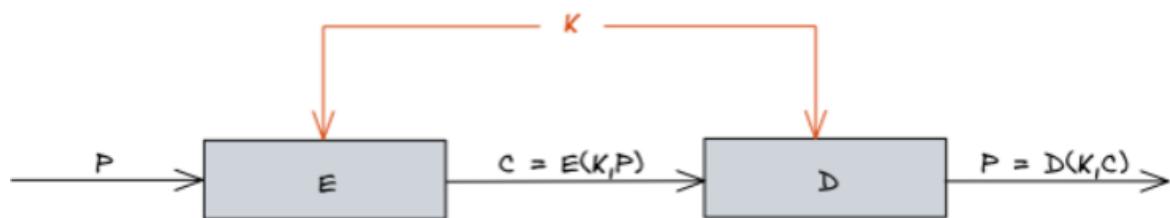
Smjer:112

Predmet: Sigurnost računala i podataka

Symmetric key cryptography univerzalna je tehnika kojom se omogućava povjerljivost i kod komunikacije (prijenos) i kod pohrane podataka.

Osnovni elementi ove tehnike su:

- PLAINTEXT (P) - originalna poruka koja se enkriptira
- ENKRIPCIJSKI ALGORITAM (E) - algoritam koji primjenjuje radnje na plaintext-u
- SECRET KEY (K) - ulaz enkripcijskog algoritma o kojem i sam algoritam i njegove radnje ovise
- CIPHERTEXT (C) - izokrenuta (enkriptirana) poruka
- DECRYPTION ALGORITHM (D) - algoritam koji pomoći ciphertext-a i sigurnosnog ključa dolazi do originalne poruke



#### NAPADI NA SIMETRIČNU ENKRIPCIJU:

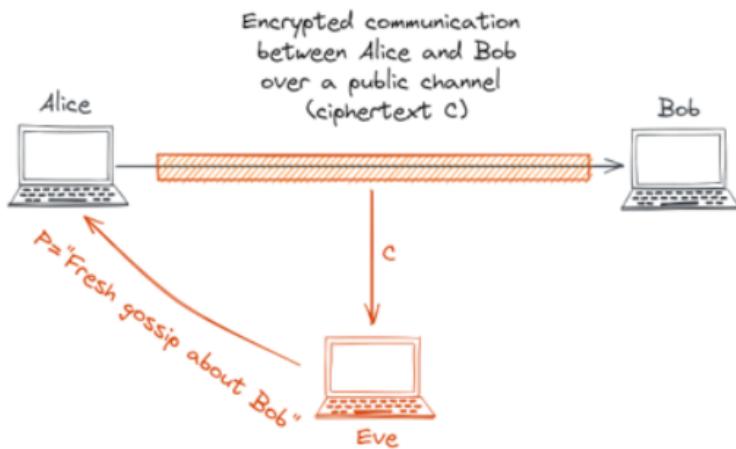
2 generalna pristupa:

→ kriptoanaliza

→ brute-force napad

Plaintext koji se treba otkriti enkriptiran je korištenjem high-level sustava za simetričnu enkripciju iz biblioteke Fernet. U ovoj se vježbi za dekriptiranje personalizirane enkriptirane poruke koristi *brute-force napad*.

*Brute-force napad* univerzalan je i izravni napad koji obuhvaća par (P,C). Postupak ovog napada obuhvaća isprobavanje svakog mogućeg ključa te uspoređivanje je li enkriptirana poruka dekriptirana pomoći odgovarajućeg ključa jednaka početnom (običnom) tekstu.



### IZVOĐENJE VJEŽBE:

1. pozicioniranje u odgovarajući direktorij
2. stvaranje direktorija koji obuhvaća sve izvršne datoteke koje bi mogle biti potrebne pri rađenju projekta u pythonu, naredba `python -m venv __(ime)`
3. naredbom `pip install cryptography` se u pozicioniranom direktoriju stvara biblioteka naziva `cryptography`
4. uključivanje biblioteke Fernet
5. služenje osnovnim naredbama gdje naredbom `generate_key()` generiramo neki sigurnosni ključ, naredbom `f.encrypt` enkriptiramo neki tekst, a naredbom `decrypt` pomoću sigurnosnog ključa dekriptiramo poruku kako bi došli do početnog teksta
6. pomoću naredbe `hash` enkriptiramo naziv datoteke, koji se sastoji od prezimena, povlake i imena studenta ,da bi našli personalizirani zadatak pod odgovarajućom enkriptiranom porukom
7. u pythonu se definira funkcija `brute_force()` gdje se datoteka sa zadatkom otvara i čita naredbama `open` i `read`, a funkcijom `while` se omogućava dekripcija na način se isprobavaju svi mogući sigurnosni ključevi dok se ne nađe ključ pomoću kojeg se može doći do plaintext-a

### ZAKLJUČAK:

Na temelju izvedene vježbe možemo uvidjeti važnost entropije u generiranju sigurnosnog ključa, pogotovo u razlici kod pronalaženja ključa 20 i 22 bita entropije. Također, naizgled neraspoznatljiv ključ ne mora podrazumijevati i veliku entropiju.

