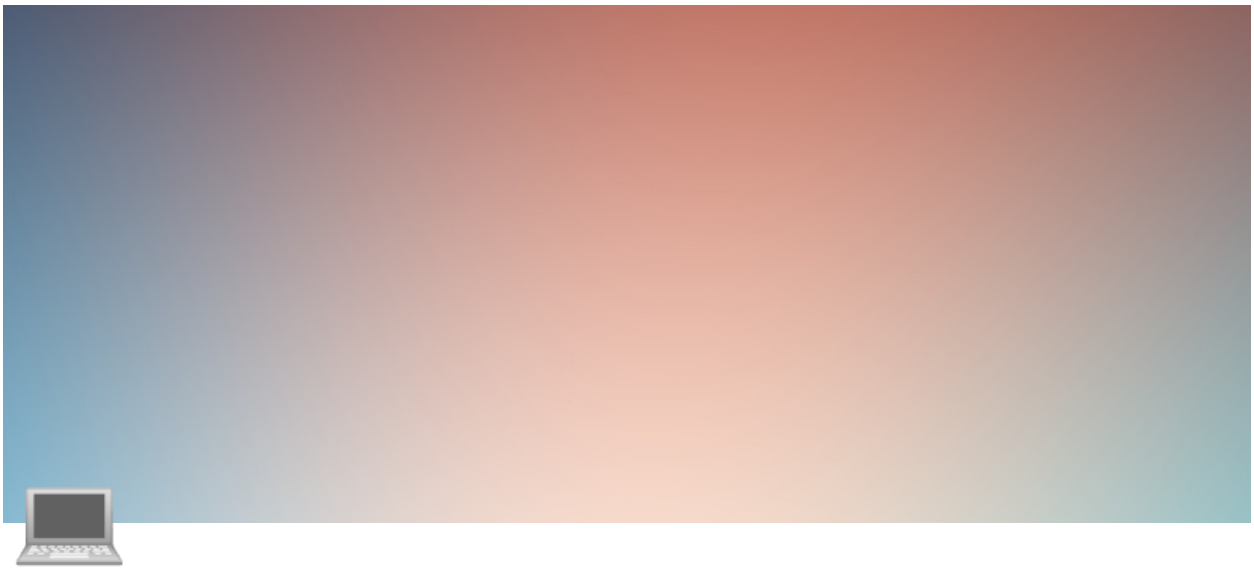

Izvještaj četvrte laboratorijske vježbe

Zadatak1: Digitalni potpis

U ovom zadatku je trebalo odrediti *autentičnu sliku* (između dvije ponuđene) koju je profesor potpisao svojim *privatnim ključem* pritom smo koristili RSA kripstosustav iz Python biblioteke *cryptography.*

Zadatak2: Password-hashing (iterative hashing, salt, memory-hard functions)

Zaporke/lozinke su najzastupljeniji način autentikacije korisnika. U okviru vježbe upoznati ćemo se pobliže sa osnovnim konceptima relevantnim za sigurnu pohranu lozinki. Usporediti ćemo klasične (*brze*) kriptografske *hash* funkcije sa specijaliziranim (*sporim* i *memorijski zahtjevnim*) kriptografskim funkcijama za sigurnu pohranu zaporki i izvođenje enkripcijskih ključeva (*key derivation function (KDF)*).

-Iterative hashing

- Insted of hashing password **p** only once and storing **H(p)**, hash **p** iteratively **n** times and store $H^n(p) = H(...H(H(p))...)$

-**Salt**

In cryptography, a salt is random data that is used as an additional input to a one-way function that hashes data or password.

-**Memory-hard functions**

A memory-hard function is a function that costs significant amount of memory to evaluate.

Provjeravali smo vrijeme izvršavanja nekoliko kripto hash funkcija. Neke su bile jako brze,pa povećavanjem brojem iteracija možemo jako usporiti napadača te tako povećamo sigurnost funkcija.