Privacy Enhancing Technologies: Intro + FHE

Florian Tramèr

Privacy Enhancing Technologies

Private computations:

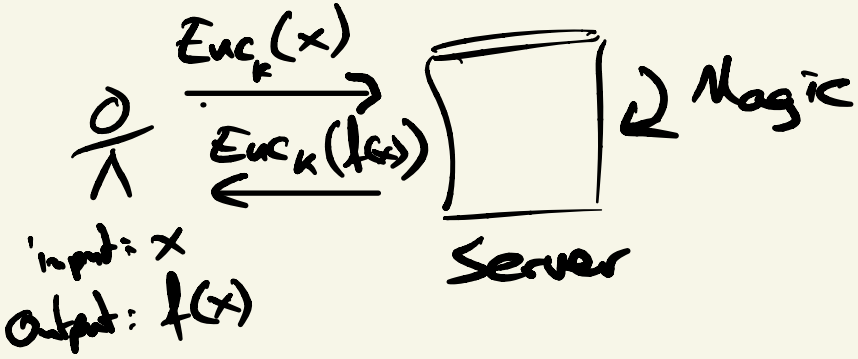
$$y_1, \ldots, y_n \leftarrow f(x_1, \ldots, x_n)$$

$$\perp, m \leftarrow f_{\text{Enc}}(m, \perp)$$

Logistics

spylab.ai/teaching/pets-f26

Fully Homomorphic Encryption (FHE)



More formally: $FHE = (Gen, Enc, Dec, Eval)$

Symmetric encryption scheme

$$c' \leftarrow Eval(f, c_1, \ldots, c_n)$$

$c_1, \ldots, c_n$: ciphertexts

$$Dec_k(Eval(f, Enc_k(x_1), \ldots, Enc_k(x_n))) = f(x_1, \ldots, x_n)$$

Obs: non-malleability / IND-CCA

"given a ciphertext c, the adversary cannot create a valid ciphertext c' s.t $Dec(c) \neq Dec(c')$"

Extra property: compactness

$|c'|$ should be indep of $|f|$

↑ output of Eval

↳ the # of ops to compute f

Obs: we can write f as a series of '+' and '×' over some field

Many schemes support **one** of these

Ex: El-Gamal $Enc(m) = (g^r, m \cdot h^r)$

$$Enc(m_1) \circ Enc(m_2) = Enc(m_1 \cdot m_2)$$
$$= (g^{r_1+r_2}, m_1 m_2 \cdot h^{r_1+r_2})$$

Somewhat Homomorphic Encryption

↳ compute a bounded # of '+' and '×' before decryption fails

Ex: $\text{Gen}()$: key = "large" odd integer p

$\text{Enc}_p(b)$: $c = p \cdot q + 2r + b$

where q is "large" random and r is "small" random

$$\text{Dec}_p(c) = ((c \bmod p) \bmod 2)$$

$\text{Eval}('+', c_1, c_2) = c_1 + c_2$

Homomorphic: $((c_1 + c_2) \bmod p) \bmod 2$

$= (2 \cdot r_1 + b_1) + (2r_2 + b_2) \bmod 2$

$= b_1 + b_2$

$((c_1 \cdot c_2) \bmod p) \bmod 2 = b_1 b_2 \bmod 2$

$\text{Eval}('\times', c_1, c_2) = c_1 \cdot c_2$