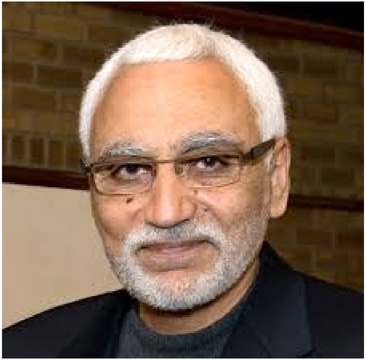
ElGamal



Taher Elgamal (born 18 August 1955, Egypt).

ElGamal encryption scheme

ElGamal encryption scheme

KeyGen

- Pick a large prime numbers p for which the **discrete logarithm** is hard and a generator g of $\mathbb{Z}_p^*$ (the group of invertible elements of $\mathbb{Z}_p$).
- Choose a random element $x \in \mathbb{Z}_{\phi(p)}$, and set $X = g^x \mod p$.
- Set $pk = (X, p, g)$ as public key, and $sk = (x)$ as secret key.

——————————————————————————–

Encryption

- Pick a random value $y \in \mathbb{Z}_{\phi(p)}$, and compute $Y = g^y \mod p$.
- The encryption of the message m is computed as $c = X^y m \mod p$.
- The final cipher text is $C = (Y, c)$.

——————————————————————————–

Decryption

- Parse the ciphertext as $C = (Y, c)$, and compute the shared secret key $K = Y^x$.
- Compute the modular inverse of K in $\mathbb{Z}_p$ (e.g. using EEA or Fermat little Theorem)
- Retrieve the plaintext by computing $m = cK^{-1} \mod p$.