



Public Key Encryption from trapdoor permutations

Constructions

Goal: construct chosen-ciphertext secure public-key encryption

Trapdoor functions (TDF)

Def: a trapdoor func. $X \rightarrow Y$ is a triple of efficient algs. (G, F, F^{-1})

- $G()$: randomized alg. outputs a key pair (pk, sk)
- $F(pk, \cdot)$: det. alg. that defines a function $X \rightarrow Y$
- $F^{-1}(sk, \cdot)$: defines a function $Y \rightarrow X$ that inverts $F(pk, \cdot)$

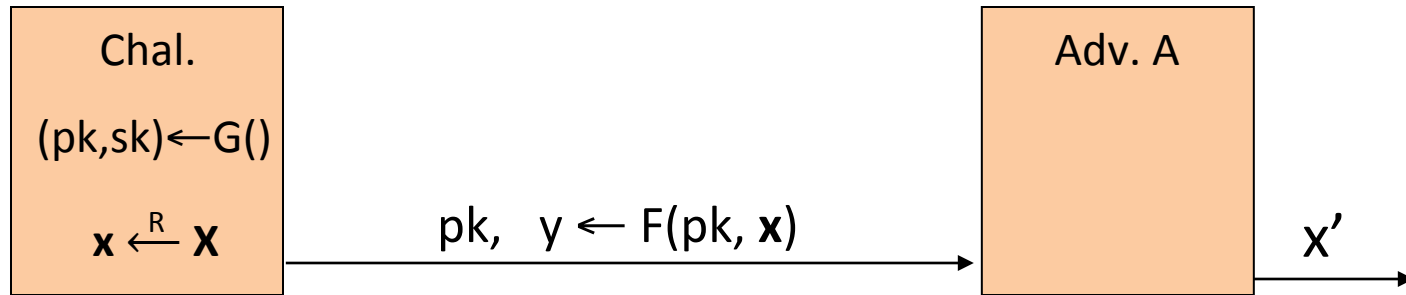
More precisely: $\forall (pk, sk)$ output by G

$$\forall x \in X: F^{-1}(sk, F(pk, x)) = x$$

Secure Trapdoor Functions (TDFs)

(G, F, F^{-1}) is secure if $F(pk, \cdot)$ is a “one-way” function:

can be evaluated, but cannot be inverted without sk



Def: (G, F, F^{-1}) is a secure TDF if for all efficient A :

$$\text{Adv}_{\text{OW}}[A, F] = \Pr[x = x'] < \text{negligible}$$

Public-key encryption from TDFs

- (G, F, F^{-1}) : secure TDF $X \rightarrow Y$
- (E_s, D_s) : symmetric auth. encryption defined over (K, M, C)
- $H: X \rightarrow K$ a hash function

We construct a pub-key enc. system (G, E, D) :

Key generation G : same as G for TDF

Public-key encryption from TDFs

- (G, F, F^{-1}) : secure TDF $X \rightarrow Y$
- (E_s, D_s) : symmetric auth. encryption defined over (K, M, C)
- $H: X \rightarrow K$ a hash function

$E(pk, m)$:

$x \xleftarrow{R} X, \quad y \leftarrow F(pk, x)$

$k \leftarrow H(x), \quad c \leftarrow E_s(k, m)$

output (y, c)

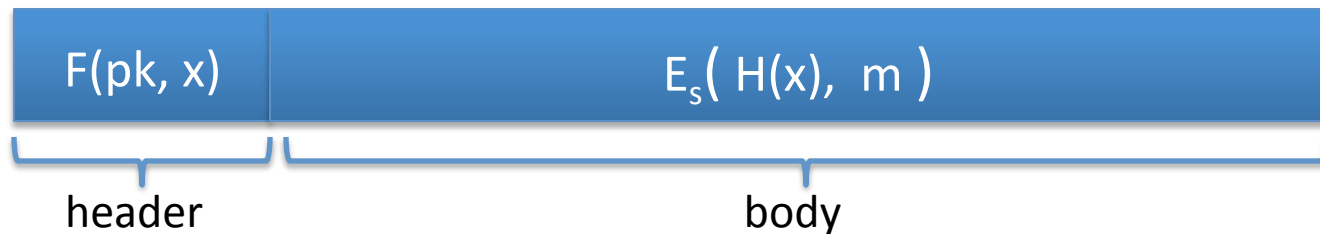
$D(sk, (y, c))$:

$x \leftarrow F^{-1}(sk, y),$

$k \leftarrow \text{[redacted]} \quad m \leftarrow D_s(k, c)$

output m

In pictures:



Security Theorem:

If (G, F, F^{-1}) is a secure TDF, (E_s, D_s) provides auth. enc.
and $H: X \rightarrow K$ is a “random oracle”
then (G, E, D) is CCA^{ro} secure.

Incorrect use of a Trapdoor Function (TDF)

Never encrypt by applying F directly to plaintext:

$E(pk, m)$:

output $c \leftarrow F(pk, m)$

$D(sk, c)$:

output $F^{-1}(sk, c)$

Problems:

- Deterministic: cannot be semantically secure !!
- Many attacks exist (next segment)

Next step: construct a TDF

End of Segment