



Probabilistic Encryption

Prof. Zeph Grunschlag

Symmetric Encryption

DEF: A **symmetric encryption scheme** consists of a tuple (M, K, G, E, D) where

- M - message space
- K - key space
- G - randomized key generator picks key k of **security parameter l** . Write: $k \stackrel{R}{\leftarrow} G(1^l)$
- E - randomized (possibly *stateful*) encryption algorithm producing ciphertext from key and plaintext. Write: $c \stackrel{R}{\leftarrow} E_k(m)$
- D - deterministic (possibly *stateful*) decryption algorithm producing plaintexts from ciphertexts s.t. $\forall m, D_k(E_k(m)) = m$

Blum-Goldwasser PKE (Asymmetric)

- $M =$ all bitstrings
- $K = \{(p, q) \mid p \neq q \text{ primes, } |p| = |q|, p \equiv q \equiv 3 \pmod{4}\}$
 - $P(k) =$ public key $= n = p \cdot q$
 - $T(k) =$ trapdoor key $= (p, q)$
- $G =$ On input 1^l :

Generate $p \neq q$, random l -bit Miller-Rabin-Pseudoprimes (of chosen certainty)

BG PKE - encryption

$E(n, m)$ // key n , message m

$L = |m|$

$r \in_R \mathbb{Z}_n$ // r picked uniformly at random

$s = \text{BBS-PRG}(n, r, L)$

$t = \text{bitstring}(r^{2^{L+1}} \bmod n)$ // keep leading 0's

return $(m \oplus s) || t$ // xor and concatenate

NOTES:

1. $|t| = |n|$ as keeping leading 0's of number
2. Could replace BBS-PRG by any PRG

Blum-Blum-Shub PRG

INPUT: key n , seed r , expansion L

OUTPUT: bitstring s of length L

BBS-PRG(n, r, L)

$x = r^2 \bmod n$ // $r \in_R \mathbb{Z}_n \Rightarrow x \in_R \text{QR}(n)$

for $i = 1$ to L {

$s_i = x \bmod 2$ // least significant bit

$x = x^2 \bmod n$ // replace by square

}

return $s_1 \parallel s_2 \parallel \dots \parallel s_L$ // concatenate bits

BG PKE - decryption

$D((p, q), c)$ // private key p, q , ciphertext c

$L = |c| - |p \cdot q|$ // subtract the length of t

$y = \text{binarynumber}(c[L+1, |c|])$ // last $|t|$ -bits

$$r_p = y^{[(p+1)/4]^{L+1}} \bmod p$$

$$r_q = y^{[(q+1)/4]^{L+1}} \bmod q$$

$$r = [q(q^{-1} \bmod p)r_p + p(p^{-1} \bmod q)r_q] \bmod n$$

$s = \text{BBS-PRG}(n, r, L)$

return $c[1, L] \oplus s$ // xor first L bits of cipher

NOTE: r above only has probability $1/4$ of being same r as during encryption; however, squares the same so is an equivalent BBS-PRG seed.

Multi-Message Distinguisher

DEF: A **multi-message distinguisher** for an encryption scheme (M, K, G, E, D) is a decision algorithm A that attempts to discover which of two chosen message-sequences

$$[m_{1,a}, m_{2,a}, \dots, m_{q,a}], [m_{1,b}, m_{2,b}, \dots, m_{q,b}]$$

a cipher-sequence $[E_k(m_{1,?}), E_k(m_{2,?}), \dots, E_k(m_{q,?})]$

corresponds to. Define the the **a-b**

advantage of A : $\text{Adv}(A, [m_{i,a}], [m_{i,b}])$

$$= \text{Prob}(A(E_k(m_{i,a})) = 1) - \text{Prob}(A(E_k(m_{i,b})) = 1)$$

Computational Security

DEF: Let (M, K, G, E, D) be an encryption scheme with G, E, D running in poly-time. The scheme is **computationally insecure** if there is a PPT multi-message distinguisher A with non-negligible a-b advantage for some equal-size messages. I.e. can find a poly-number of messages $m_{i,a}, m_{i,b}$ with $|m_{i,a}| = |m_{i,b}|$ but $\text{Adv}(A, m_{i,a}, m_{i,b})$ non-negligible.

- Measure time/space/negligibility in terms of security parameter l for the key generator.
- Say **computationally secure** if **not** computationally insecure - no such A exists.

Adaptive Chosen Plaintext Attack Secure

- Looked at carefully, previous definition is for Chosen Plaintext Attack (CPA) because insecure if *some* distinguishable message sequences exist, so these could be *chosen*.
- Adaptive CPA: Can modify the chosen messages as go along. Would modify security definition to allow adversary A to have oracle access to a counter-adversary C which when given chosen messages $(m_{i,a}, m_{i,b})$ returns $E_k(m_{i,?})$ for a consistent unknown “?”

Non-negligible Function

DEF: A function $f : \mathbb{N} \rightarrow \mathbb{R}$ is **non-negligible** if there is a polynomial $p(n)$ such that

$$|f(n)| = \Omega\left(\frac{1}{p(n)}\right)$$

Stateless Deterministic Encryption

THM: Any stateless, deterministic encryption is insecure. In fact, there is an adversary A with advantage 1 for some well chosen message sequences.

NOTE: One-time-pad avoids this problem because under this paradigm, there is an implicit counter whose value > 1 implies encryption is refused and the output “ \perp ” is returned for any plaintext.

Weaker Notions

Consider instead **ciphertext security**:

DEF: The cryptosystem (M, K, G, E, D) is NOT ciphertext-secure under chosen plaintext attack if there is a PPT cryptanalysis algorithm P and some polynomial number of plaintexts which when given as input the known $[m_i]$ ciphertexts $[E_k(m_i)]$ and unknown ciphertext c , P returns the plaintext m corresponding to c with non-negligible probability for a non-negligible fraction of ciphertexts c .

LEMMA: adaptive-CPA computational security implies ciphertext security.