Part II of Introduction to Cryptography: Probabilistic Encryption

Konstantin Beznosov
COT 6421 / Spring 1998

April 2, 1998

We Will Discuss Today:

- Why do we need probabilistic encryption?
- The idea behind
- Optimized algorithm
- Drawbacks

Why do we need probabilistic encryption?

$$C = E_k(M)$$

$C' = E_k(M')$ and C'=C $\Rightarrow$ M' = M

The idea behind probabilistic encryption

$$C_1 = E_k(M),\ C_2 = E_k(M),\ C_3 = E_k(M), ..., C_i = E_k(M)$$

$$M = D_k(C_1) = D_k(C_2) = D_k(C_3) = ...\, D_k(C_i)$$

$C_i = E_k(M)$ even if M' = M it cannot be checked by comparing $C_i = E_k(M)$ $C_j = E_k(M')$

Simplified description of the optimized algorithm

p and q are primes

p=3 mod 4 and q = 3 mod 4

private key – p and q

public key – n=pq

Optimized algorithm: encryption

1. Choose some random *x*, relatively prime to *n*.

2. Compute $x_0 = x^2\, mod\, n$

3. Run BBS generator with x_0 as the seed. The generator spits out bits b_i, where each b_i is the least significant bit of $x_i \equiv x_{i-1}^2 mod\, n$

4. Use the output of the generator as a stream cipher.

5. Compute XOR M, one bit at a time, with the output of the generator.

   M=$m_1, m_2, m_3, ..., m_t$ C=$m_1 \oplus b_1, m_2 \oplus b_2, m_3 \oplus b_3, ..., m_t \oplus b_t$

6. Append the last computed value, x_t, to the end of the message C.

Decryption & Drawbacks

Decryption

Values of p,q,n, t and x_t are used to recover x_0 and the original plaintext.

Drawbacks of probabilistic encryption:

- Ciphertext large size
- Totally insecure against a chosen-ciphertext attack