

Solutions

ENEE 459E/CMSC 498R: Introduction to Cryptology RSA Cryptanalysis

3. Sending the same message to multiple receivers:

The following is a slightly extended version of Chinese Remainder Theorem than the one we saw in class for the case where there are 3 moduli.

Chinese Remainder Theorem. Let N_1, N_2, N_3 be pairwise relatively prime. Then for every c_1, c_2, c_3 , there exists a unique non-negative integer \hat{c} such that:

$$\begin{aligned}\hat{c} &= c_1 \bmod N_1 \\ \hat{c} &= c_2 \bmod N_2 \\ \hat{c} &= c_3 \bmod N_3.\end{aligned} \rightarrow \hat{c} < N^3$$

Assume there are three receivers with public keys:

$$pk_1 = \langle N_1, 3 \rangle, pk_2 = \langle N_2, 3 \rangle, pk_3 = \langle N_3, 3 \rangle.$$

A sender sends the same encrypted message m to all three receivers so an eavesdropper sees:

$$c_1 = m^3 \bmod N_1, c_2 = m^3 \bmod N_2, c_3 = m^3 \bmod N_3$$

Show how to use the Chinese Remainder Theorem to recover m .

Since the same message m is encrypted 3 times, we have that

$$\begin{aligned}m^3 &= c_1 \bmod N_1 \\ m^3 &= c_2 \bmod N_2 \\ m^3 &= c_3 \bmod N_3\end{aligned}$$

given c_1, c_2, c_3 we can use CRT to find

$$\hat{c} < N^3 \text{ such that } \begin{aligned}\hat{c} &= c_1 \bmod N_1 \\ \hat{c} &= c_2 \bmod N_2 \\ \hat{c} &= c_3 \bmod N_3\end{aligned}$$

since \hat{c} is unique, we must have $\hat{c} = m^3$

can now take cube root of \hat{c} over the integers
(which is easy - your calculator can do it!)
to recover m .

Solutions

ENEE 459E/CMSC 498R: Introduction to Cryptology
RSA Cryptanalysis
5/2/17

1. Partially Known Message.

Coppersmith's Theorem: Let $p(x)$ be a polynomial of degree e . Then in time $\text{poly}(\log(N), e)$ one can find all m such that $p(m) = 0 \pmod{N}$ and $m \leq N^{1/e}$.

Assume message is $m = m_1 || m_2$, where m_1 is known, but m_2 (which consists of k bits) is not known. Using Coppersmith's Theorem, show how to recover m given the ciphertext c , # of bits in N assuming k is not too large.

Hint: Note that m can be expressed as $m := 2^k m_1 + m_2$. if $k < \frac{\|N\|}{e} = \frac{\log N}{e}$

Set polynomial $p(m_2) := (2^k m_1 + m_2)^e - c = 0 \pmod{N}$

This is a polynomial of degree e .

Using Coppersmith can solve for m_2 satisfying the above

as long as $m_2 \leq N^{1/e}$. Thus, we check that this condition holds:

$$m_2 < 2^k < 2^{\|N\|/e} = 2^{\log N/e} = N^{1/e} \quad \checkmark$$

2. Related Messages.

Euclidean Algorithm for Polynomials: Let $f(x)$ and $g(x)$ be two polynomials over Z_N^* . Then a slightly modified version of the Euclidean GCD Algorithm can be used to determine the greatest common divisor of f, g as polynomials over Z_N^* .

Assume the sender encrypts both m and $m + \delta$, for known δ , unknown m giving two ciphertexts c_1 and c_2 . Use the Euclidean algorithm for polynomials to show how to recover m given knowledge of δ and given the two ciphertexts c_1, c_2 .

$$\begin{aligned} \text{Set } f(x) &:= x^e - c_1 \pmod{N} \\ g(x) &:= (x + \delta)^e - c_2 \pmod{N} \end{aligned}$$

Note that when $x = m$, we have $g(m) = f(m) = 0$.

Therefore $(x - m)$ is a root of both $g(x)$ and $f(x)$.

Using gcd for polynomials we can find $x - m$, thus determining m .