HW #2 - CNS Sapienza

Daniele Oriana 1457625

12 Nov 2015

20150114-Q2

RSA
Q2.1 [3/30] Describe how RSA encryption/decryption work.
Q2.2 [2/30] Explain what is the mathematical relationship between e and d (the two exponents used in RSA).
Q2.3 [3/30] Why the textbook implementation of RSA is insecure? Provide at least an example.

Answer

Q2.1

Rsa is an asymmetric encryption algorithm that was published in 1977. This algorithm derives directly from the cryptographic direction taken by Diffie-Helmann in 1976, using the same principle of key exchange. The innovation introduced by Rsa provides for division of the old single key of an entity (used in the previous algorithms) into two new keys: one used to encrypt the messages going to the entity (Kp) and the other one used by the entity in question to decrypt received messages (ks).
Kp is made public, achieving, in this way, an algorithm based on asymmetric and public key. The procedure to apply the algorithm provides the following steps:

- Choose two distinct prime number p and q and compute their product: N=pq
- Choose one number $1<e<\Phi(N)$, such that $\gcd(e , \Phi(N)) = 1$, note that $\Phi(N)$ is the Eulero's function that indicates the number of coprime of N between 1 and N, while $\gcd(e , \Phi(N))$ refers to the greatest common divisor

of e and $\Phi(N)$, from this follows that if gcd(e , $\Phi(N)$) = 1 then e and $\Phi(N)$ are coprime
- Choose a number d such that de= 1 mod $\Phi(N)$, this means that d is the multiplicative inverse of e
- The public key is the pair (e , N)
- The private key is the pair (d , N)

After this procedure we can encrypt a message M doing: C= E(M)= M^e mod N, while we can decrypt a ciphertext C computing: M= D(C)= C^d mod N.
So anyone can send a message to an involved entity encrypting through its public key, but only the receiving entity can decrypt using its private key.

Q2.2

The whole procedure of RSA, already described, works thanks to some special mathematical properties of the exponents e and d, in fact, from the previous calculations, we obtain:

$$C^d = (M^e)^d = M^{ed} \bmod N$$

but, according with the previous decision taken during the choice of d, we have that

ed = 1 mod (p-1)(q-1) and consequently ed = 1 mod (p-1) and ed = 1 mod (q-1).

So, by the Fermat's little theorem:

$$M^{ed} = M \bmod p \text{ and } M^{ed} = M \bmod q$$

and, since p and q are two different prime number, we can apply the Chinese remainder theorem

$M^{ed} = M \bmod (pq)$ and so we have that $C^d = M \bmod N$.

The Fermat's little theorem provides that if p is a prime number, then for every integer a we have that $a^p = a \bmod p$, while the Chinese remainder theorem states that if we consider a sequence of positive integers n1, n2, , nk which are pairwise coprime, then for any given integers a1, a2, , ak there exists an integer x solving the system of simultaneous congruencies

$x = a_i \bmod n_i$ for i = 1, , k.

Q2.3

The textbook implementation of RSA can be broken by different attacks, ad example by a chosen chipertext attack. We suppose to have the usual situation with Alice, Bob and Trudy (the attacker). The chosen chipertext attack for RSA can be described by the following steps:
- Alice sends to Bob $C = E(M) = M^e \bmod N$
- Trudy intercepts the message and changes it with $C_b = C_a * C$, where $C_a = 2^e \bmod N$, so $C_b = (2M)^e \bmod N$
- Trudy sends C_b to Bob that computes $(C^b)^d = 2M$

So the attacker can obtain the original message dividing by two the value calculating by Bob. Note that Bob has received a message that is different from the original.