

## Agenda

**PROBLEM DOMAIN: NUMBER THEORY**

**-APPLICATION DOMAIN: CRYPTOGRAPHY**

**- PUBLIC KEY ENCRYPTION : RSA  
SECURITY OF RSA**

# RSA: Security

- Protocol RSA: (Public Key Encryption)
  - Offline Steps (i.e. pre-processing) by B or a third party trusted by B:
    - Choose  $n = p * q$  for large primes  $p$  and  $q$
    - Choose  $k$  in  $Z_{\phi(n)}^*$  and publish  $(n, k)$  as public key for B
    - Compute B's private key  $k'$ , the inverse of  $k$  in  $(Z_{\phi(n)}^*, * \phi(n))$
  - Online Steps (i.e. at communication time):
    - Let  $E(M) = M^k \pmod{n}$
    - A sends  $M' = E(M)$  to B
    - B receives  $M'$
    - B computes  $E^{-1}(M') = (M')^{k'} \pmod{n}$
- Security Requirement:
  - Given  $n$  (but not  $p$  nor  $q$ ), and  $k$ 
    - an attacker cannot get  $M$  in polynomial time

# RSA: Security

- (Provable) Security Correctness Requirement:
  - Given  $n$  (but not  $p$  nor  $q$ ), and  $k$ 
    - an attacker cannot get any of these:
      - $p$  or  $q$ 
        - because factoring is “hard”
      - $\phi(n)$ 
        - because computing  $\phi(n)$  is as “hard” as factoring
      - $k'$ , the inverse of  $k$  in  $(\mathbb{Z}_{\phi(n)}^*, *)$ 
        - because if  $k'$  is known, then the attacker knows  $\phi(n) \mid (k * k' - 1)$
        - i.e. attacker knows  $j * \phi(n)$  for some +ve integer  $j$ 
          - *Then  $n$  can be factorized efficiently.*
          - *(claim w/o proof).*

# RSA: Security - Pragmatics

- (Provable) Security Property for RSA:
  - Given  $n$  (but not  $p$  nor  $q$ ), and  $k$ 
    - an attacker cannot get any of these:
      - $p$  or  $q$
      - $\phi(n)$
      - $k'$ , the inverse of  $k$  in  $(\mathbb{Z}_{\phi(n)}^*, *n)$
- The above statement states the hardness of breaking RSA scheme completely - by computing the private key
  - Alternatively, is there a way to infer (i.e. decrypt) messages without the decryption key?
  - i.e. we want a guarantee of the form:
    - *It is not possible to decode more than a small fraction of encrypted messages*

# RSA: Security - Pragmatics

- We want a guarantee of the form:
  - *It is not possible to decode more than a small fraction of encrypted messages*
- Given an attacker's algorithm  $A$  that knows only  $n$  and  $k$ , define
  - $C(A) = \{x \text{ in } Z_n^* \mid A \text{ can compute } x^{k'} \pmod{n} \text{ given } x\}$ 
    - where  $k'$  is the inverse of  $k$  in  $(Z_{\phi(n)}^*, *n)$
  - i.e.  $C(A)$  is the set of messages in  $Z_n^*$  that can be recovered using  $A$ .

# RSA: Security - Pragmatics

- Theorem:
  - Suppose there exists a (possibly randomized) polynomial time algorithm  $A_1$  for which  $|C(A_1)| \geq \epsilon * |Z_n^*|$  for some  $\epsilon > 0$ .
  - Then there exists a Las Vegas algorithm  $A_2$  for which
    - $|C(A_2)| = |Z_n^*|$  and
    - the expected running time of  $A_2$  is polynomial in  $\log(n)$  and  $1/\epsilon$ .
- Implications:
  - If RSA can be broken (i.e. a more than a small number of messages decrypted) then it can be broken almost completely:
    - Note that  $|Z_n^*| = p * q + 1 - (p + q) = \Theta(|Z_n|)$ 
      - for  $n = p * q$  where  $p$  and  $q$  are large primes.
  - If  $\epsilon$  is vanishingly small, say for instance  $o(1/n)$ , then the expected time complexity is exponential in size of  $n$ 
    - i.e. the attack – using  $A_2$  – may not be practical.