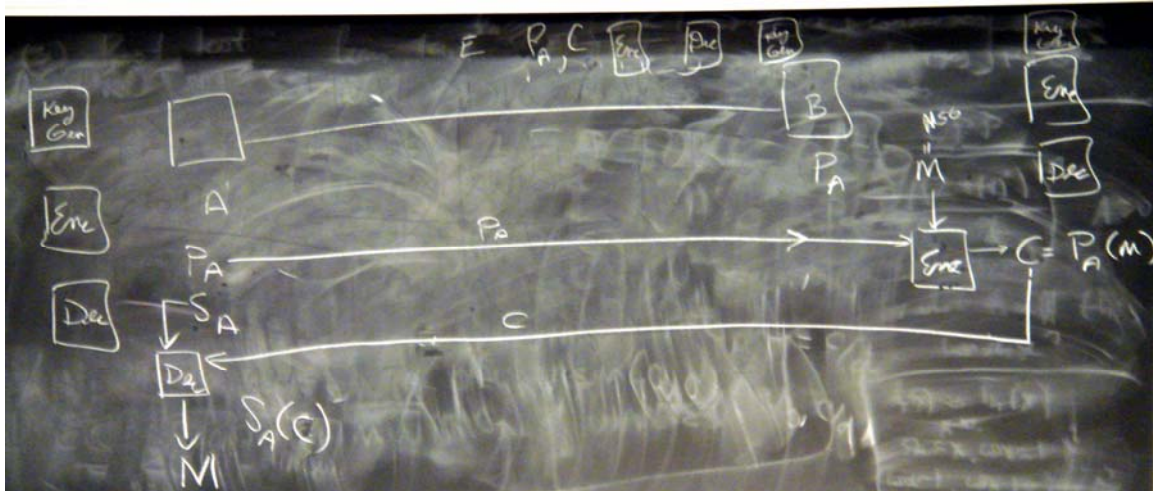
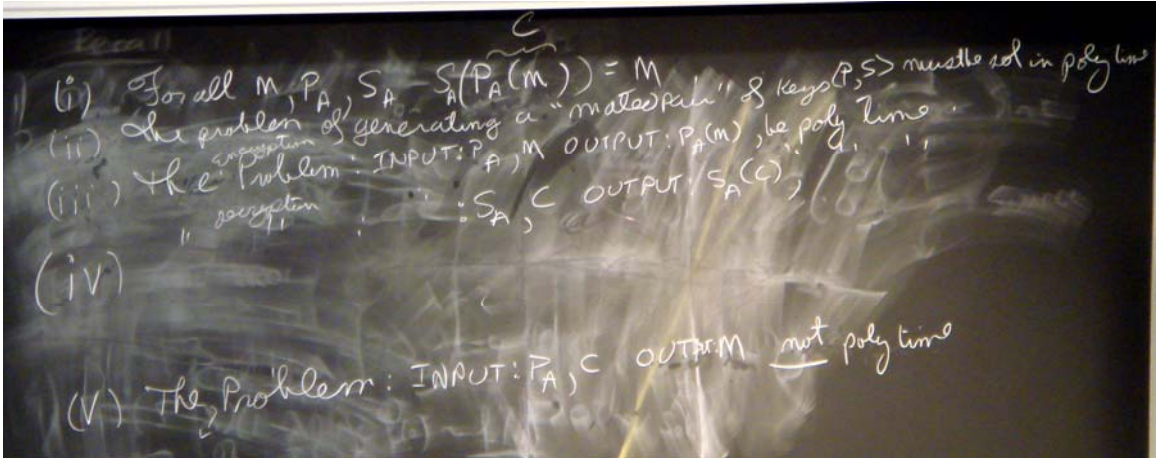


Bob wants to send Alice a message:



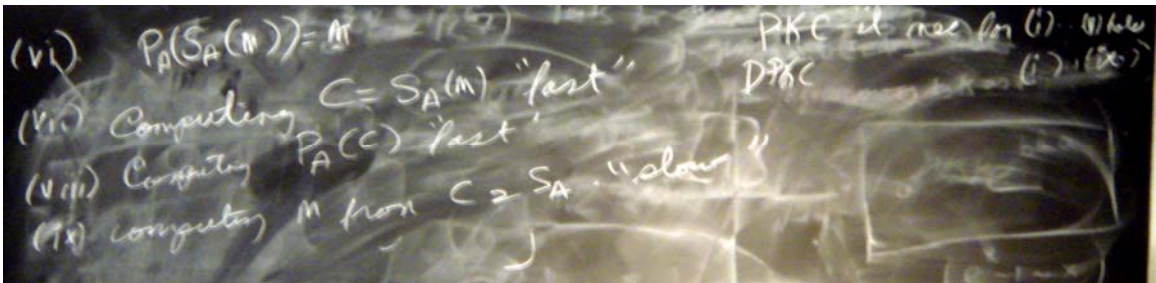
Requirements for a PKC (Public Key Cryptosystem):

- i) For all m , $P_A, S_A, S_A(P_A(m)) = m$.
- ii) The problem of generating a “mated pair” of keys $\langle P, S \rangle$ must be solvable in polynomial time.
- iii) The encryption problem:
 Input: P_A, m
 Output: $P_A(m)$
 must be solvable in polynomial time.
- iv) The decryption problem:
 Input: S_A, c
 Output: $S_A(c)$
 must be solvable in polynomial time.
- v) The code breaking problem:
 Input: P_A, c
 Output: m
 should not be solvable in polynomial time.



Additionally, for DPKC (Double Public Key Cryptosystems), where the public and secret keys are invertible:

- vi) For all $m, P_A, S_A, P_A(S_A(m)) = m$.
- vii) The problem:
 - Input: S_A, m
 - Output: $S_A(m)$
 - must be solvable in polynomial time.
- viii) The problem:
 - Input: P_A, c
 - Output: $P_A(c)$
 - must be solvable in polynomial time.
- ix) The problem:
 - Input: S_A, c
 - Output: m
 - should not be solvable in polynomial time.



RSA:

Generating a “mated pair” of keys.

1. Generate two large (1024 bits) primes, p and q .
2. Compute $n = pq$, and $\phi(n) = (p - 1)(q - 1)$.
3. Generate e such that $\text{GCD}(e, \phi(n)) = 1$.
4. Compute d such that $ed \equiv 1 \pmod{\phi(n)}$.
5. Public key $PK = \langle e, n \rangle$
6. Secret key $SK = \langle d, n \rangle$

Encryption:

Compute $PK(m) = m^e \pmod{n}$

Decryption:

Compute $SK(c) = c^d \pmod{n}$



Example:

1. $p = 5, q = 11$
 2. $n = 5(11) = 55, \phi(n) = (5 - 1)(11 - 1) = 40$
- ...short aside

The fundamental theorem of arithmetic:

For all natural numbers n , there exists a unique prime factorization of n .

Factoring problem:

Input: n

Output: $S = \{\langle p_1, e_1 \rangle, \langle p_2, e_2 \rangle, \dots, \langle p_w, e_w \rangle\}$ where all p_i are prime and e_i are positive

integers such that $\prod_{i=1}^w p_i^{e_i} = n$.

To date, we do not know of a polynomial time algorithm for the factoring problem; however, we have also not been able to show that it is NP-complete.

GCD: greatest common divisor. It can be found using Euclid's algorithm in polynomial time.