

## Exercise Sheet 4

Out: Tue, Dec 15, 2009

Due: Tue, Jan 5, 2010, 10am

**Problem 1: Computational implementations (10 points)  
for public key encryption**

Let  $(K, E, D)$  be an IND-CCA secure encryption scheme with probabilistic polynomial-time key-generation  $K$  and encryption  $E$ , and deterministic polynomial-time decryption  $D$ . Assume that for all  $m \in \{0, 1\}^*$ ,  $k \in \mathbb{N}$ ,

$$\Pr[m = m' : (ek, dk) \leftarrow K(1^k), c \leftarrow E(ek, m), m' \leftarrow D(dk, c)] = 1. \quad (1)$$

(That is, decryption never fails.)

Construct a computational implementation  $A$  for the symbolic model  $\mathbf{M}_{pke}$  (as in Section 8.1 in the lecture notes) that satisfies the implementation conditions given in Definition 44 in the lecture notes.

**Note:** Do not assume anything about the encryption scheme except what is explicitly given in the problem statement. In particular, do not assume that all encryption/verification keys have the same length, that only  $k$ -bits of randomness are used, that the encryption key can be efficiently extracted from the ciphertext, ...

**Problem 2: A complete analysis (10 points)**

Consider the following protocol:

- Protocol participants are Alice and Bob. They communicate over an insecure channel (i.e., all messages are effectively sent to and received from the adversary).
- Let  $(ek_B, dk_B)$  be a preshared encryption key pair. Assume that Alice knows  $ek_B$  and Bob knows  $dk_B$ . (The actual distribution of these keys is not part of the protocol.)
- Alice chooses a nonce  $N_A$ .
- Alice encrypts  $N_A$  using  $ek_B$  and sends the resulting ciphertext  $c$  to Bob.
- Bob raises an event *begin* and then decrypts  $c$ , extracts the nonce  $N_A$  and sends  $N_A$  to Alice.
- When Alice receives  $N_A$ , she raises an event *end*.

We say the protocol is secure if the event *end* never occurs unless the event *begin* has occurred before.

- (a) Model the above protocol as a CoSP protocol  $\Pi$ . Model the security of the protocol (i.e., *end* only occurs after *begin*) as a trace property  $\wp$ .

**Note:** Make sure that you do not implicitly assume some kind of synchronization (i.e., that the first step of Alice occurs before the first step of Bob and the first step of Bob occurs before the second step of Alice).

- (b) Show that  $\Pi$  symbolically satisfies  $\wp$ . You may do this either by hand or using a theorem prover like SPASS (depending on your preferences and mood).
- (c) Show that  $\Pi$  computationally satisfies  $\wp$ , i.e., that the protocol described above is secure even in a cryptographic sense.

**Hint:** Use Lemma 9 from the lecture notes.