Foundations of Cryptography

Oded Goldreich and Shafi Goldwasser

Exercise 3 (rev2): Zero knowledge, encryption and a bit of number theory

Due date: 28 Feb. 2002

1. In class we saw a a zero-knowledge interactive proof for 3-colorability, which is NP-complete. We will see that this yields a zero-knowledge interactive proof for any language in NP. Recall that $L \in NP$ iff there exists a relation R_L such that $L = \{x : \exists y (x, y) \in R_L\}$ and membership of (x, y) in R_L is decidable in time poly($|x|$).

   (a) Given an interactive proof for 3-colorability, for each $L \in NP$ obtain an interactive proof for L. Do this so that if the 3-colorability prover can be implemented in probabilistic polynomial-time when given a 3-coloring of the input graph, then the prover for L can be implemented in probabilistic polynomial-time when given an adequate NP-witness.

   Hint: The generic reduction of L to 3-colorability is coupled with a polynomial-time transformation of NP-witnesses (i.e., of witness for membership in L into a 3-coloring).

   (b) Show that the transformation can be made to preserve (auxiliary input) zero-knowledge. That is, given a simulator for the 3-colorability prover, construct a simulator for the prover derived for L.

   Hint: Be careful about the inputs that are available to the cheating verifier V^*. You will need to use one of the following facts:

      i. The 3-colorability prover is auxiliary-input zero-knowledge (rather than merely zero-knowledge).

      ii. The generic reduction of L to 3-colorability is invertible in polynomial-time.

2. Recall that a public key cryptosystem (G, E, D) has indistinguishability security (for multiple messages) if for every polynomial-size circuit family $\{A_k\}_k$, every polynomial $t(k)$, all sufficiently large k and all $\bar{m}_0 = (m_{0,1}, \ldots, m_{0,t(k)})$, $\bar{m}_1 = (m_{1,1}, \ldots, m_{1,t(k)})$ where $m_{i,j} \in \{0,1\}^k$:

$$\left| \Pr\left[A_k(e, c_1, \ldots, c_{t(k)}) = 1 \middle| (e,d) \leftarrow G(1^k)\,;\, c_i \leftarrow E_e(m_{0,i})\right] - \Pr\left[A_k(e, c_1, \ldots, c_{t(k)}) = 1 \middle| (e,d) \leftarrow G(1^k)\,;\, c_i \leftarrow E_e(m_{1,i})\right] \right| < \mathrm{negl}(k)$$

   (a) Show that if there exists a public-key cryptosystem that has indistinguishability security then there exist one-way functions.

   (b) Show that if there exists a public-key cryptosystem that has indistinguishability security then there exist collections of trapdoor predicates.

3. For each of the following public key cryptosystems, prove one of the following claims (the strongest you can):

   - (Weak negative) A counterexample: assuming the existence of the objects (e.g., PRFs) used in the construction, show that there exist choices of these objects (e.g., some specific PRF) for which indistinguishability security is violated.
   - (Strong negative) Indistinguishability security is necessarily violated, for any choice of the objects.
   - (Strong positive) Indistinguishability security necessarily holds, for any choice of the objects.

   The cryptosystems are as follows.

   (a) Demetrius noted that in the standard $\mathrm{EPE}_{\mathrm{RSA}}$ construction, the ciphertext (which is of size $2k$) is twice as large as the plaintext (which is of size k). This seems like a terrible waste, so he propoposes to modify $\mathrm{EPE}_{\mathrm{RSA}}$ as follows. Let E be the $\mathrm{EPE}_{\mathrm{RSA}}$ encryption algorithm, and denote by E^{α} its invocation with the random value r fixed to α. Then the modified scheme encrypts messages m of size $2k$ by dividing them into equal parts $m = m_1 \parallel m_2$ and outputting $E^{m_1}_{(e,n)}(m_2)$. This lets us send twice as much information using the same ciphertext size.
   Explain how decryption works and evaluate the security of the scheme.

   (b) Demetrius, whose hand-eye coordination is not as keen as his cryptographic adventurousness, is experiencing difficulties when asked to flip too many coins. He thus proposes the following cryptosystem:
   Let $\mathcal{F} = \{F_n\}_n$ be an ensemble of one-way trapdoor permutations whose indexing, evaluation and inversion algorithms are I, V and N respectively. Let $n = k + \log k$.

   $G(1^k) : (i, t) \leftarrow I(1^n)$, where i is the index of the permutation and t is the trapdoor. i serves as the public key and t serves as the private key.

   $E_i(m)$: Let $r \in_R \{0,1\}^{\log k}$. Compute $c \leftarrow V(i, r \parallel m) = f_i(r \parallel m)$ and output c.

   $D_{i,t}(c)$: Compute $(r \parallel m) \leftarrow N(t, c) = f_i^{-1}(c)$ and output m.

   Evaluate the security of this scheme.

   (c) Demetrius gave his best friend Phila a birthday present: a public key cryptosystem (G, E, D) with indistinguishability security. Phila, who was a bit suspicious, decided to add a bit of randomness "just in case", and thus defined the following scheme:
   Let Q be a pseudorandom generator with expansion function $l(n) = 2n$.

$$
\begin{aligned}
G'(1^k) &\equiv G(1^k) \\
E'_e(m) &: \text{Let } r \in_R \{0,1\}^{\frac{k}{2}}. \\
&\quad\ \text{Output } c = (E_e(r), Q(r) \oplus m). \\
D'_d(c_1, c_2) &: \text{Compute } r \leftarrow D_d(c_1);\ \text{Output } Q(r) \oplus c_2.
\end{aligned}
$$

   Evaluate the security of Phila's scheme.

4. The Decision Diffie Hellman (DDH) hardness assumption is defined as follows:
Let $p = 2q + 1$ be a prime such that q is also prime. Let g be the *square* of a generator of $\mathbb{Z}_p^*$. Then the distributions (g^x, g^y, g^{xy}) and (g^x, g^y, g^r) for $x, y, r \in_R \{1, \ldots, p-1\}$ cannot be efficiently distinguished with probability better than $\frac{1}{2} + \text{negl}(|p|)$.

   The Computational Diffie Hellman (CDH) hardness assumption is defined as follows:
Let $p = 2q + 1$ be a prime such that q is also prime. Let g be the *square* of a generator of $\mathbb{Z}_p^*$. Then the function $(g^x, g^y) \mapsto g^{xy}$ cannot be efficiently computed with more than negligible probability of success, for $x, y \in_R \{1, \ldots, p-1\}$.

   The El-Gamal public-key encryption scheme is defined as follows:

$$
\begin{array}{lrl}
 & G(1^k): & \text{Choose arbitrary } p \text{ and } g,\ p \text{ prime, such that } p = 2q+1 \text{ for} \\
 & & \text{some prime } q, \text{ and } g \text{ is the square of a generator of } \mathbb{Z}_p^*. \\
 & & \text{Let } d \in_R \{1, \ldots, p-1\}, h = g^d \pmod{p}. \\
 & & \text{The public key is } (p, g, h) \text{ and the private key is } d. \\
\text{for } m \in QR_p: & E_{(p,g,h)}(m): & \text{Let } r \in_R \{1, \ldots, p-1\}, c_1 = g^r \pmod{p},\ c_2 = h^r m \pmod{p}. \\
 & & \text{Output } c = (c_1, c_2). \\
\text{for } c_1, c_2 \in \mathbb{Z}_p^*: & D_{(p,g,h),d}(c_1, c_2): & \text{Compute } t = {c_1}^d \pmod{p}. \\
 & & \text{Output } t^{-1} c_2 \pmod{p}.
\end{array}
$$

   QR_p denotes the set of squares mod p. Note that p and g need not be chosen at random.

   (a) Suppose we modified the DDH hardness assumption by requiring g to be a generator of $\mathbb{Z}_p^*$ rather than the square of a generator. Show that the modified assumption is false.

   Hint: Consider the Legendre symbol, i.e., squareness mod p.

   (b) Does the El-Gamal scheme have indistiguishability security under the (unmodified) DDH hardness assumption? Prove or give a counterexample.

   (c) What can you say about the security of El-Gamal under the CDH hardness assumption?

   (d) Is the El-Gamal scheme secure against chosen ciphertext attack?

5. Recall that $\mathcal{F}$ is an ensemble of pseudorandom permutations if $\mathcal{F} = \{F_n\}_n$, $F_n = \{f_i\}_i$, f_i are permuations on $\{0,1\}^n$ and the following hold:

   - There exists a p.p.t. indexing algorithm I s.t. $I(1^n) \to i$
   - There exists a p.p.t. evaluation algorithm V s.t. $V(i, x) = f_i(x)$ for all $x \in \{0,1\}^n$
   - There exists a p.p.t. inversion algorithm N s.t. $N(i, f_i(x)) = x$ for all $x \in \{0,1\}^n$
   - For any non-uniform p.p.t. distinguisher A,

$$\xi_{\mathcal{F},A}(n) = \left| \Pr\left[A^{f_i, f_i^{-1}}(1^n) = 1 \middle| i \leftarrow I(1^n)\right] - \Pr\left[A^{g, g^{-1}}(1^n) = 1\right] \right| < \text{negl}(n)$$

     where g is a random permutation on $\{0,1\}^n$.

Given an ensemble of pseudorandom permutations $\mathcal{F}$ and corresponding algorithms (I, V, N), consider the following private-key encryption scheme (G, E, D):

$$\begin{array}{ll} & G(1^k): \; s \leftarrow I(1^{2k}) \\ \text{for } m \in \{0,1\}^k: & E_s(m): \; c \leftarrow V(s, (r \parallel m)) \text{ where } r \in_R \{0,1\}^k \\ \text{for } c \in \{0,1\}^{2k}: & D_s(c): \; (r \parallel m) \leftarrow N(s,c); \text{ return } m \end{array}$$

("$\parallel$" denotes string concatenation)

(a) Let M be a non-uniform p.p.t. algorithm that performs a chosen-plaintext attack on (G, E, D) in the setting described by indistinguishibility security for a single message. Namely, M makes queries to an encryption oracle and then tries to distinguish $E_e(m_0)$ from $E_e(m_1)$ for some m_0, m_1.
Suppose M runs in time $t(k)$, makes $q(k)$ encryption queries and succeeds with probability $\frac{1}{2} + \varepsilon(k)$. Give a nonuniform distinguisher for $\mathcal{F}$ that runs in time $O(t(k))$, performs $q(k) + 1$ oracle queries and has the best $\xi_{\mathcal{F},A}(n) > 0$ you can achieve (as a function of ε).

(b) Show that (G, E, D) has indistinguishability security for multiple messages (defined as in Question 2, except that for private-key encryption scheme, A_k doesn't get the input e).

6. The best known factoring algorithm runs in time $e^{O\left((\log n)^{\frac{1}{3}} (\log \log n)^{\frac{2}{3}}\right)}$, which depends on the size of n but not on the sizes of its prime factors (unlike some less efficient algorithms, such as the trivial trial division algorithm). This suggests that in order to increase the security of RSA, it may suffice to increase just one of the prime factors.

Consider the following variant of RSA, called unbalanced RSA (or RSA for paranoids). It is identical to standard RSA, except:

- The prime p is 500 bits long and the prime q is 4500 bits long.
- The messages space is $\{0, \ldots, p-1\}$.

We may hope that unbalanced RSA is about as hard to break as standard RSA with 5000-bit n, while (as we shall see) the complexity of decryption and decryption is similar to that of standard RSA with 500-bit p and q.

(a) How would you choose the public exponent e?
RSA is often used with fixed $e = 3$. Is this a good choice for unbalanced RSA?

Decryption is done by computing $m = c^d \pmod{n}$, where $d = e^{-1} \pmod{\phi(n)}$ was computed in advance. However, exponentiation to a large power d modulu a large integer n is an expensive operation. We make the following optimization: compute $m' = c^d \pmod{p}$. For $m \in \{0, \ldots, p-1\}$ we have $m' = m$.

(b) Show how with a single chosen message attack (i.e. obtaining the decryption of a message of your choice) you can completely break the optimized unbalanced RSA scheme, by factoring n.

Questions may be directed to Shafi Goldwasser <shafi@wisdom.weizmann.ac.il> or Eran Tromer <tromer@wisdom.weizmann.ac.il>.