0368.4162 Foundations of Cryptography December 14, 2017

Problem Set 4

Due: December 28

Instructions:

- Type your solution and email it to "focf17hw@gmail.com", with subject "HW#, ID#, Name".
- You're encouraged to use latex. You can find on the webpage the assignment's tex file to help you.
- You can cooperate. However, you should write the solution by yourself, and list all collaborators for each question. Same goes for any external sources that you may use.
- Do not discuss solutions over the course's forum. You are more than welcome though to ask for clarifications regarding the questions themselves.

1. In this question, we will understand the relations between the different security notions of encryption. In what follows, let (G, E, D) be an encryption scheme. To present the definitions in a way that applies to both the secret and the public settings we shall agree that $G(1^n)$ outputs three keys (sk, ek, pk):

   - sk is the secret decryption key.
   - ek is an encryption key.
   - pk is a public key.
   - In the secret key setting, $ek = sk$ and pk is empty, whereas in the public-key setting $ek = pk$.

   We consider the following security notions:

   Known Plaintext Attacks: The scheme is t-message KPA-secure, for a polynomially bounded function t, if any n.u. PPT adveraary $A = \{A_n\}$ wins the following game with probability at most $1/2 + \mu(n)$ for some negligible μ.

   - A_n submits $t(n)$ pairs of equal length messages $(m_{0,1}, m_{1,1}), \ldots, (m_{0,t}, m_{1,t})$.
   - A_n obtains $pk, Enc_{ek}(m_{b,1}), \ldots, Enc_{ek}(m_{b,t})$ for a random $b \leftarrow \{0,1\}$.
   - A_n outputs a bit b', and wins if $b = b'$.

   Chosen Plaintext Attacks: The scheme is CPA-secure, if any n.u. PPT adveraary $A = \{A_n\}$ wins the following game with probability at most $1/2 + \mu(n)$ for some negligible μ.

   - A_n obtains pk.
   - A_n can repeatedly submit a message m and obtain an encryption $E_{ek}(m)$.
   - A_n submits two equal-length messages m_0, m_1 and obtains $Enc_{ek}(m_b)$ for a random $b \leftarrow \{0,1\}$.
   - A_n can once more repeatedly submit a message m and obtain an encryption $E_{ek}(m)$.
   - A_n outputs a bit b', and wins if $b = b'$.

   Show that:

   (a) (20 pts) Any scheme that is CPA-secure is also t-message KPA-secure for any polynomial t (in either the secret-key or public-key setting).

   (b) (**Bonus:** 10 pts) If there exist secret-key encryption schemes that are t-message KPA-secure for any polynomial t, than there exist ones that are not CPA secure.

   (c) (20 pts) Any public-key bit-encryption scheme that is 1-message KPA-secure is CPA secure.

2. In this question, we'll construct a two-message bit commitment scheme based on PRGs. Let $G : \{0,1\}^n \to \{0,1\}^{3n}$ be a length-tripling PRG. Consider the following protocol between a sender S and a receiver R:

   - R samples a random string $r \leftarrow \{0,1\}^{3n}$ and sends it to S.
   - S samples $s \leftarrow \{0,1\}^n$, and commits to a bit b, by sending $Com_r(b;s)$ defined by:
     - $G(s)$ if $b = 0$,
     - $G(s) \oplus r$ if $b = 1$.

   Show that the commitment is:

   - (15 pts) **Binding:** with overwhelming probability over the choice of r, the commitment is binding:

$$\Pr_{r \leftarrow \{0,1\}^{3n}} [\exists s_0, s_1 : Com_r(0;s_0) = Com_r(1;s_1)] \leq 2^{-n} \ .$$

   - (15 pts) **Hiding:** for every (even malicious) choice of r, the sender's message is computationally-hiding:

$$\{Com_r(0;U_n)\}_{\substack{n\in\mathbb{N} \\ r\in\{0,1\}^{3n}}} \approx_c \{Com_r(1;U_n)\}_{\substack{n\in\mathbb{N} \\ r\in\{0,1\}^{3n}}} \ .$$

3. (30 pts) Consider the NP language of solvable Soduko puzzles. That is, all $n^2 \times n^2$ matrices over values $\{\perp\} \cup [n^2]$ for which the values $\perp$ can be replaced with values in $[n^2]$, so that every line , column, or $n \times n$ subsquare contain the values $[n^2]$. (By subsquare here we mean the entries corresponding to the intersection of rows $in+1, \ldots, in+n$ with columns $in+1, \ldots, in+n$ for some $i, j \in \{0, \ldots, n-1\}$.)

   Describe a zero-knowledge proof for this language, based on non-interactive commitments, with perfect completeness, soundness error $s \leq 1 - n^{-O(1)}$, and expected poly-time simulator. Re zero-knowledge, you can describe the simulator, without writing the proof of its validity and run-time.