Handout 2: Perfect and Statistical Secrecy, Computational Models

Boaz Barak

Exercises due Tuesday September 27, 2005 1:30pm
Total of 130 points.

1 Suggested Reading

Perfect secrecy is covered in Bellare's and Golwasser-Bellare lecture notes (see the web site for links). Computational models such as Boolean circuits and probabilistic computation are covered in Sipser's book pages 251–260 (Boolean circuits) and 368–375 (probabilistic algorithms).

2 Perfect security and statistical closeness

Exercise 1 (15 points)**.** Show formally that the following schemes do *not* satisfy the definition of perfect security (you can choose the most convenient of the 3 equivalent definitions to demonstrate this).

Notation: we use $\mathbb{Z}_n$ to denote the set of numbers $\{0, \ldots, 26-1\}$. We identify the letters of the English alphabet with $\mathbb{Z}_{26}$ in the obvious way. We denote by ℓ the length of the plaintext p.

1. Caesar cipher (Key: a random $k \leftarrow_{\mathrm{R}} \mathbb{Z}_{26}$. Encryption: $E_k(p_1, \ldots, p_\ell) = (p_1 + k, \ldots, p_\ell + k) \pmod{26}$)
2. Substitution cipher (key: a random permutation $\pi : \mathbb{Z}_{26} \to \mathbb{Z}_{26}$, encryption: $E_\pi(p_1, \ldots, p_\ell) = (\pi(p_1), \ldots, \pi(p_\ell)$.
3. "Two-time pad". Key: $\leftarrow_{\mathrm{R}} \{0,1\}^{\ell/2}$ (assume plaintext are over Binary alphabet and of even length). Encryption $E_k(p_1, \ldots, p_\ell) = (p_1 \oplus k_1, \ldots, p_{\ell/2} \oplus k_{\ell/2}, p_{\ell/2+1} \oplus k_1, \ldots, p_\ell \oplus k_{\ell/2})$.

Exercise 2 (15 points)**.** Describe (in words, without proofs) how you could extend the one-time pad into an encryption with similar security that allows to encrypt *multiple messages* with the same key, as long as the total length of all the messages is less than the length of the key.

You are allowed to use a *stateful encryption*. That is, both the encryption algorithm and decryption algorithm can maintain a state variable/register that is updated from encryption to encryption. Is it possible to do so without using state for the decryption algorithm? what about the encryption algorithm?

Exercise 3 (20 points)**.** Give examples (with proofs) for

1. A scheme such that it is possible to efficiently recover 90% of the bits of the key given the ciphertext, and yet it is still perfectly secure. Do you think there is a security issue in using such a scheme in practice?

2. An encryption scheme that is *insecure* but yet it provably hides the first 20% bits of the key. That is, if the key is of length n then the probability that a computationally unbounded adversary guesses the first $n/5$ bits of the key is at most $2^{-n/5}$.

You can use the results proven in class and above. Also the examples need not be natural schemes but can be "contrived" schemes specifically tailored to obtain a counter-example.

Exercise 4 (20 points)**.** Recall that we defined the statistical distance of X and Y, $\Delta(X,Y)$ as follows: let U denote the union of the supports of X and Y. For every set $T \subseteq U$, denote $\Delta_T(X,Y) = |\Pr[X \in T] - \Pr[Y \in T]|$. Then $\Delta(X,Y) = \max_T \Delta_T(X,Y)$. Prove that

1. $\Delta(X,Y) = \frac{1}{2}\sum_{u \in U} |\Pr[X=u] - \Pr[Y=u]|$
2. The statistical distance satisfies a triangle inequality/ transitivity property. That is, for any three random variables X, Y, Z,

$$\Delta(X,Z) \leq \Delta(X,Y) + \Delta(Y,Z)$$

Exercise 5 (10 points)**.** Recall that a scheme is ϵ statistically indistinguishable if in the distinguishing between two messages game, the probability adversary guesses the messages is at most $\frac{1}{2} + \epsilon$. A scheme satisfies ϵ-statistical secrecy if for every x, x' the distributions Y_x and $Y_{x'}$ (defined to be $E_{U_n}(x)$ and $E_{U_n}(x')$ respectively) satisfy $\Delta(Y_x, Y_{x'}) \leq \epsilon$.

Prove that a scheme (E,D) is ϵ statistically indistinguishable if and only if it satisfies 2ϵ-statistical secrecy.

Exercise 6 (20 points)**.** Suppose that (D,E) is a valid encryption scheme with key 100 bit shorter than the plaintext. That is, $E : \{0,1\}^n \times \{0,1\}^{n+100} \to \{0,1\}^m$ for some $n, m \in \mathbb{N}$. Show that the scheme can be completely broken by an explicit (although not necessarily efficient) *algorithm*. That is, show that there exist two messages x_1, x_2 and an algorithm Adv (not necessarily efficient) such that

$$\Pr_{i \in \{1,2\}\,,\, k \leftarrow_R \{0,1\}^n}[\mathsf{Adv}(E_k(x_i)) = i] > 0.99$$

In rough terms (up to polynomial accuracy), what is the running time of your algorithm?
Can you find a stronger attack if the message is of size $100n$?
Suppose that $\mathbf{P} = \mathbf{NP}$. Can you use that to find a faster algorithm?

Exercise 7 (10 points)**.** Read the handout on computational models. Write down one question you want to know the answer for and is not explained (or not explained clearly) in the handout.

Exercise 8 (20 points)**.** Consider the following three decision problems/languages:

- CSAT - Circuit Satisfiability. **Input:** a Boolean circuit C with n inputs and one output. **Output:** 1 iff there exists $x \in \{0,1\}^n$ such that $C(x) = 1$.
- SAT - satisfiability. **Input:** a 3CNF formula ψ with inputs $x_1, \ldots, x_n$. A 3CNF formula is a formula of the form $C_1 \wedge C_2 \wedge \cdots \wedge C_m$ where each of the C_i's (called clauses) is a disjunction $(l_1 \vee l_2 \vee l_3)$. Each of the l_j's (called literals) is either a variable x_k or a negation of a variable $\neg x_{k'}$. **Output:** 1 iff there exists inputs $x_1, \ldots, x_n$ that satisfy the formula.
- $\mathsf{3COL}$ - 3 colorability. **Input:** an undirected graph $G = (V,E)$ with n vertices. **Output:** 1 iff there exists a valid 3-coloring of G. A 3-coloring is a function $c : V \to \{1,2,3\}$ such that for every $(u,v) \in E$, $c(u) \neq c(v)$.

1. Prove that all three problems are in **NP** (this should be easy and take at most 1-2 lines for each problem)

2. CSAT is **NP**-complete. Use that to prove that SAT is **NP** complete. (Hint: the formula does not have to use the exact same number of variables as the circuit - use additional auxiliary variables that represent the circuit's internal gates.)

3. 3COL is **NP**-complete. (Hint: use the previous item.) Note that we'll actually use the fact that 3COL is **NP**-complete (much) later in the course.

Note: These are classical results and appear in many textbooks (including Sipser). Do not look at the proofs of these results while solving this problem. (If you came across these results earlier in a book/another course this is fine, but you should not consult the book/your notes while solving this problem and writing its solution.)

What is the **NP**-completeness status of 2COL? (Same as 3COL but with 2 instead of 3 colors, give a short 1-3 line answer).