6.875J/18.425J Cryptography and Cryptanalysis	October 24, 2001
Handout 12: Problem Set 7	
Instructor: Anna Lysyanskaya	*Teaching Assistant: Moses Liskov*

This problem set is due on Wednesday, November 21.

Problem 1: Random Functions and Random Permutations

Let $\mathcal{O}_0 = R(n)$ be a random (length-preserving) function oracle: on any input from $\{0,1\}^n$, if it has not yet given an answer for that input, it returns a random value from $\{0,1\}^n$.

Let $\mathcal{O}_1 = \Pi(n)$ be a random permutation oracle: on any input from $\{0,1\}^n$, if it has not yet given an answer for that input, it returns a random value in $\{0,1\}^n$ that it has never returned before.

Prove that $\mathcal{O}_0(n)$ is indistinguishable from $\mathcal{O}_1(n)$, in the strongest sense you can.

Problem 2: Attacking Luby-Rackoff

Recall the Luby-Rackoff construction for a pseudorandom permutation from a pseudorandom function. Let F_i be a set of pseudorandom functions, and let

$$LR_i(L,R) = (R, F_i(R) \oplus L)$$

define the i^{th} round of operation of the Luby-Rackoff construction. For an integer k, define

$$LR - k(L,R) = LR_k(LR_{k-1}(\dots(L,R)\dots)$$

to be the output of k-round Luby-Rackoff.

We say a permutation family P_S over $\{0,1\}^{l(k)}$ is pseudorandom if:

$\forall \{A_k^?\} \in PPT\ \exists \nu \in negl\ \forall k$
$\Pr[S \leftarrow \{0,1\}^k; \mathcal{O}_0 \leftarrow P_S;$
$\quad \mathcal{O}_1 \leftarrow \Pi(l(k));$
$\quad b' \leftarrow A_k^{\mathcal{O}_b} : b' = b] < 1/2 + \nu(k)$

(a) Prove that $LR - 2$ is not a pseudorandom permutation family.

We say a permutation family P_S is strongly pseudorandom if:

$\forall \{A_n^{?,?}\} \in PPT\ \exists \nu \in negl\ \forall k$
$\Pr[S \leftarrow \{0,1\}^k; \mathcal{O}_0 \leftarrow P_S; \mathcal{O}'_0 \leftarrow P_S^{-1}$
$\quad \mathcal{O}_1 \leftarrow \Pi; \mathcal{O}'_1 \leftarrow \Pi^{-1};$
$\quad b' \leftarrow A^{\mathcal{O}_b, \mathcal{O}'_b} : b' = b] < 1/2 + \nu(k)$

where Π^{-1} inverts Π. Think of Π as being a fixed but randomly chosen perumtation, and Π^{-1} as its inverse.

(b) Prove that $LR-3$ is not a strong pseudorandom permutation family.

Problem 3: Four-Round Luby-Rackoff

In class we proved that $LR-3$ is a pseudorandom permutation family. Prove that $LR-4$ is a strong pseudorandom permutation family. This is rather difficult to come up with from scratch, so we provide a proof sketch that you should fill in.

Proof sketch: First of all, assume we have an adversary $A^{?,?}$ which defeats the security of $LR-4$. Prove that without loss of generality we can assume that the adversary never asks $\mathcal{O}$ or $\mathcal{O}'$ the same question twice, that the adversary never queries $\mathcal{O}$ on an output of $\mathcal{O}'$, and that the adversary never queries $\mathcal{O}'$ on the output of $\mathcal{O}$.

Next, formulate a construction for $LR-4R$, which performs the Luby-Rackoff function with 4 random functions instead of F_1, F_2, F_3, F_4. This construction is just like the one from class, except for $LR-4$ instead of $LR-3$. Prove that $LR-4R$ is indistinguishable from a random function, similarly to how we did in class.

Next, formulate a construction for $(LR-4R)^{-1}$, which operates with the oracle $LR-4R$, and prove that adversaries that have oracle access to $LR-4R$ and another (challenge) oracle cannot tell whether the challenge oracle was $(LR-4R)^{-1}$ or a random function. This is the meat of the proof.

Finally, put it all together and prove that $LR-4$ is indistinguishable from a random permutation in this model. Your proof should assume this is not the case and derive a contradiction by proving that one of the functions F_i is not pseudorandom.

Problem 4: Encrypting the Secret Key

Let (G, E, D) be a (one-pass) GM-secure cryptosystem. This should be thought of as a "secure envelope" – the adversary cannot see its contents, no matter what the message is. This problem shows that one has to be careful with this intuition. Namely, we ask the question, "Can I trust my encryption to the extent that I can publish the encryption of my secret key?"

(a) (7 points) Construct a cryptosystem (G', E', D') based on (G, E, D) such that given $t = E'(PK', SK')$, the adversary can easily distinguish the encryptions of *any* two messages m_0 and m_1. (Hint: t should somehow contain SK'. Try to imagine how this can be done without making (G', E', D') insecure.)

(b) (7 points) Construct a cryptosystem (G'', E'', D'') based on (G, E, D) such that even giving $t = E''(PK'', SK'')$ to the adversary, the scheme is still GM-secure. You should imagine that the adversary learns t at the same time as PK''. (Hint: t should somehow contain random garbage. Try to imagine how we can do this but *still be able to decrypt* t)

(c) (6 points) Suppose Alice and Bob each have keys, (PK_A, SK_A) and (PK_B, SK_B) under (G, E, D). Prove that Alice can securely encrypt *her* secret key under *Bob's* public key.

That is, prove that (G, E, D) is still GM-secure even when the adversary knows PK_A, PK_B, and $E(PK_B, SK_A)$. Formally, prove that

$$\begin{aligned}
&\forall \{A_k\} \in PPT\ \exists \nu \in negl\ \forall k \\
&\Pr[(PK_A, SK_A) \leftarrow G(1^k); (PK_B, SK_B) \leftarrow G(1^k); t \leftarrow E(PK_B, SK_A); \\
&\quad (\alpha, m_0, m_1) \leftarrow A(PK_A, PK_B, t); b \leftarrow \{0,1\}; b' \leftarrow A(\alpha, E(PK_A, m_b)) : b' = b] < 1/2 + \nu(k)
\end{aligned}$$

Problem 5: Insecure Signatures

All of the signature schemes presented in this problem are insecure. A signature scheme can be broken in a number of ways: it can be shown to be existentially forgeable, it can be target-message forgeable, or its secret key can be completely recoverable. In order to break a signature scheme, different flavors of attacks can be launched: a public-key only attack, or an interactive attack where the adversary asks for signatures on messages of his choice.

In this problem, you are asked to break each in the strongest sense possible, using the weakest attack you can.

On the other hand, each of these signature schemes is secure in a weak sense: namely, on input a public key and a message, it is infeasible to compute the signature. Prove this weak security property for each signature scheme.

(a) (8 points) Suppose a family of trapdoor permutations $\{p_{PK}\}$ over $\{0,1\}^k$ is given. The message space is $\{0,1\}^k$. A public key is the public key PK of a member of the TDP; the secret key is the correspoding SK. The signature on message m is the value x such that $p_{PK}(x) = m$.

(b) (6 points) Assume that RSA is a family of trapdoor permutations. A signature scheme is as follows: the public key is an RSA public key (n, e), and the secret key is the corresponding RSA secret key d. The message space is $\mathbb{Z}_n^*$. The signature on message m is the value $m^d \bmod n$.

(c) (6 points) Assume that factoring is hard. The public key of this signature scheme consists of a Blum integer $n = pq$. (Recall that Blum integers have the property that -1 is a quadratic non-residue with Jacobi symbol 1). The secret key is n's factorization. The message space is $J_1(n)$, i.e. the set of elements of $\mathbb{Z}_n^*$ that have Jacobi symbol 1. A signature σ on message $m \in J_1(n)$ is computed as follows: if m is a qudratic residue, then σ is some arbitrary square root of m. Otherwise, σ is some arbitrary square root of $-m$.