U.C. Berkeley — CS276: Cryptography Handout N3
Luca Trevisan January 27, 2009

Notes for Lecture 3

Scribed by Bharath Ramsundar, posted January 30, 2009

Summary

Last time we introduced the setting of *one-time symmetric key encryption*, defined the notion of *semantic security*, and proved its equivalence to *message indistinguishability*.

Today we complete the proof of equivalence (found in the notes for last class), discuss the notion of *pseudorandom generator*, and see that it is precisely the primitive that is needed in order to have message-indistinguishable (and hence semantically secure) one-time encryption. Finally, we shall introduce the basic definition of security for protocols which send multiple messages with the same key.

1 Pseudorandom Generators And One-Time Encryption

Intuitively, a Pseudorandom Generator is a function that takes a short random string and stretches it to a longer string which is almost random, in the sense that reasonably complex algorithms cannot differentiate the new string from truly random strings with more than negligible probability.

Definition 1 (Pseudorandom Generator) *A function $G : \{0,1\}^k \to \{0,1\}^m$ is a (t,ϵ)-secure pseudorandom generator if for every boolean function T of complexity at most t we have*

$$|\mathbb{P}_{x \sim U_k}[T(G(x)) = 1] - \mathbb{P}_{x \sim U_m}[T(x) = 1]| \leq \epsilon \tag{1}$$

(We use the notation U_n for the uniform distribution over $\{0,1\}^n$.)

The definition is interesting when $m > k$ (otherwise the generator can simply output the first m bits of the input, and satisfy the definition with $\epsilon = 0$ and arbitrarily large t). Typical parameters we may be interested in are $k = 128$, $m = 2^{20}$, $t = 2^{60}$ and $\epsilon = 2^{-40}$, that is we want k to be very small, m to be large, t to be huge, and ϵ to be tiny. There are some unavoidable trade-offs between these parameters.

Lemma 2 *If $G : \{0,1\}^k \to \{0,1\}^m$ is $(t, 2^{-k-1})$ pseudorandom with $t = O(m)$, then $k \geq m - 1$.*

Proof: Pick an arbitrary $y \in \{0,1\}^k$. Define

$$T_y(x) = 1 \Leftrightarrow x = G(y)$$

It is clear that we may implement T with an algorithm of complexity $O(m)$: all this algorithm has to do is store the value of $G(y)$ (which takes space $O(m)$) and compare its input to the stored value (which takes time $O(m)$) for total complexity of $O(m)$. Now, note that

$$\mathbb{P}_{x \sim U_k}[T(G(x)) = 1] \geq \frac{1}{2^k}$$

since $G(x) = G(y)$ at least when $x = y$. Similarly, note that $\mathbb{P}_{x \sim U_m}[T(x) = 1] = \frac{1}{2^m}$ since $T(x) = 1$ only when $x = G(y)$. Now, by the pseudorandomness of G, we have that $\frac{1}{2^k} - \frac{1}{2^m} \leq \frac{1}{2^{k+1}}$. With some rearranging, this expression implies that

$$\frac{1}{2^{k+1}} \leq \frac{1}{2^m}$$

which then implies $m \leq k + 1$ and consequently $k \geq m - 1$ □

Exercise 1 *Prove that if $G : \{0,1\}^k \to \{0,1\}^m$ is (t, ϵ) pseudorandom, and $k < m$, then*

$$t \cdot \frac{1}{\epsilon} \leq O(m \cdot 2^k)$$

Suppose we have a pseudorandom generator as above. Consider the following encryption scheme:

- Given a key $K \in \{0,1\}^k$ and a message $M \in \{0,1\}^m$,

$$Enc(K, M) := M \oplus G(K)$$

- Given a ciphertext $C \in \{0,1\}^m$ and a key $K \in \{0,1\}^k$,

$$Dec(K, C) = C \oplus G(K)$$

(The XOR operation is applied bit-wise.)

It's clear by construction that the encryption scheme is correct. Regarding the security, we have

Lemma 3 *If G is (t,ϵ)-pseudorandom, then (Enc, Dec) as defined above is $(t-m, 2\epsilon)$-message indistinguishable for one-time encryption.*

Proof: Suppose that G is not $(t-m, 2\epsilon)$-message indistinguishable for one-time encryption. Then $\exists$ messages M_1, M_2 and $\exists$ algorithm T of complexity at most $t-m$ such that

$$|\mathbb{P}_{K\sim U_k}[T(Enc(K, M_1)) = 1] - \mathbb{P}_{K\sim U_k}[T(Enc(K, M_2)) = 1]| > 2\epsilon$$

By using the definition of Enc we obtain

$$|\mathbb{P}_{K\sim U_k}[T(G(K) \oplus M_1)) = 1] - \mathbb{P}_{K\sim U_k}[T(G(K) \oplus M_2)) = 1]| > 2\epsilon$$

Now, we can add and subtract the term $\mathbb{P}_{R\sim U_m}[T(R) = 1]$ and use the triangle inequality to obtain that $|\mathbb{P}_{K\sim U_k}[T(G(K) \oplus M_1) = 1] - \mathbb{P}_{R\sim U_m}[T(R) = 1]|$ added to $|\mathbb{P}_{R\sim U_m}[T(R) = 1] - \mathbb{P}_{K\sim U_k}[T(G(K) \oplus M_2) = 1]|$ is greater than 2ϵ. At least one of the two terms in the previous expression must be greater that ϵ. Suppose without loss of generality that the first term is greater than ϵ

$$|\mathbb{P}_{K\sim U_k}[T(G(K) \oplus M_1)) = 1] - \mathbb{P}_{R\sim U_m}[T(R) = 1]| > \epsilon$$

Now define $T'(X) = T(X \oplus M_1)$. Then since $H(X) = X \oplus M_1$ is a bijection, $\mathbb{P}_{R\sim U_m}[T'(R) = 1] = \mathbb{P}_{R\sim U_m}[T(R) = 1]$. Consequently,

$$|\mathbb{P}_{K\sim U_k}[T'(G(K)) = 1] - \mathbb{P}_{R\sim U_m}[T'(R) = 1]| > \epsilon$$

Thus, since the complexity of T is at most $t-m$ and T' is T plus an xor operation (which takes time m), T' is of complexity at most t. Thus, G is not (t,ϵ)-pseudorandom since there exists an algorithm T' of complexity at most t that can distinguish between G's output and random strings with probability greater than ϵ. Contradiction. Thus (Enc, Dec) is $(t-m, 2\epsilon)$-message indistinguishable. □

2 Security for Multiple Encryptions: Plain Version

In the real world, we often need to send more than just one message. Consequently, we have to create new definitions of security for such situations, where we use the same key to send multiple messages. There are in fact multiple possible definitions of security in this scenario. Today we shall only introduce the simplest definition.

Definition 4 (Message indistinguishability for multiple encryptions) (Enc, Dec) *is* (t, ϵ)*-message indistinguishable for* c *encryptions if for every* $2c$ *messages* $M_1, \ldots, M_c$, $M'_1, \ldots, M'_c$ *and every* T *of complexity* $\leq t$ *we have*

$$|\mathbb{P}[T(Enc(K, M_1), \ldots, Enc(K, M_c)) = 1] \\ -\mathbb{P}[T(Enc(K, M'_1), \ldots, Enc(K, M'_c)) = 1]| \leq \epsilon$$

Similarly, we define semantic security, and the asymptotic versions.

Exercise 2 *Prove that no encryption scheme* (Enc, Dec) *in which* $Enc()$ *is deterministic (such as the scheme for one-time encryption described above) can be secure even for 2 encryptions.*

Encryption in some versions of Microsoft Office is deterministic and thus fails to satisfy this definition. (This is just a symptom of bigger problems; the schemes in those versions of Office are considered completely broken.)

If we allow the encryption algorithm to keep *state* information, then a pseudorandom generator is sufficient to meet this definition. Indeed, usually pseudorandom generators designed for such applications, including RC4, are optimized for this kind of "stateful multiple encryption."

Next time, we shall consider a stronger model of multiple message security which will be secure against *Chosen Plaintext Attacks*.