CS195-7: Introduction to Cryptography October 15-17, 2002

Lectures 11-12

Instructor:Anna Lysyanskaya

These lecture notes contain the proof of the Goldreich-Levin theorem.

1 Recap of definitions

1.1 One-way functions and one-way and trapdoor permutations

Definition 1.1. *A set of efficient probabilistic algorithms* $(G(\cdot), S(\cdot), f.(\cdot))$*, specifies a family of one-way functions (OWF family) if for all PPTF* $\{A_k\}$ *there exists a negligible function* $\nu(k)$ *such that*

$$\Pr[PK \leftarrow G(1^k); x \leftarrow S(PK); y \leftarrow f_{PK}(x); x' \leftarrow A_k(PK, y) : f(x) = f(x')] \leq \nu(k)$$

and for all $PK \in G$*, for all* $x \in S(PK)$*,* f_{PK} *is a function.*

Definition 1.2. *A family of one-way functions* (G, S, p) *is a family of one-way permutations (OWP family) if for all* $PK \in G$*,* p_{PK} *is a permutation.*

Definition 1.3. *A family of one-way permutations* (G, S, p) *is trapdoor (TDP family) if (1) the output of* G *consists of two values, denoted* (PK, SK)*; and (2) there is an efficient algorithm* D *such that for all* k*,*

$$\Pr[(PK, SK) \leftarrow G(1^k); x \leftarrow S(PK); x' \leftarrow D(SK, p_{PK}(x)) : x = x'] = 1$$

Definition 1.4. *A family of one-way functions (resp., permutations)* (G, S, f) *is a one-way function (resp., permutation) if* G *is a deterministic algorithm.*

1.2 Hardcore bit

Motivation: How to encrypt using a trapdoor permutation family, such as RSA. How to construct pseudorandom sequences.

Definition 1.5. *Let* (G, S, f) *be a OWF family. Let* $b_{(\circ)}(\circ)$ *be a Boolean function.* b *is a hardcore bit for* (G, S, f) *if*

1. $b_{PK}(x)$ *is approximately a bit of information about* x*. More precisely, there exists a negligible function* μ *such that for all* $PK \in G(1^k)$

$$|\Pr[x \leftarrow S(PK); b_{PK}(x)] - 1/2| \leq \mu(k)$$

2. *It is hard to compute* $b_{PK}(x)$ *from the values* PK *and* $f_{PK}(x)$*. More precisely, for all PPTF* $\{A_k\}$ *there exists a negligible function* $\nu(k)$ *such that*

$$\Pr[PK \leftarrow G(1^k); x \leftarrow S(PK); b' \leftarrow A_k(PK, f_{PK}(x)) : b' = b_{PK}(x)] \leq 1/2 + \nu(k)$$

Note that although a hardcore bit is defined for OWF families, this concept makes much more sense when we consider OWP families.

Given a hardcore bit, we can build:

- Secure PK cryptosystem: given PK, to encrypt $m \in 0,1$, pick x such that $b_{PK}(x) = m$, send $f_{PK}(x)$.
- Psudorandom generator (convert short string to long string) (see next lecture.) This allows us to increase complexity of PK cryptosystem.

2 The Goldreich-Levin theorem

Theorem 2.1. *For any OWP family, for all but a negligible fraction of $r \in \{0,1\}^{\ell}$, for all but a negligible fraction of permutations in the OWP family, $b_r(x) \equiv x.r \equiv \bigoplus_{i=1}^{\ell} x_i r_i$ is hardcore.*

This is a rather technical proof. The reason that I want to show it to you is that I believe it is very important. In some sense, this theorem says that if there is any hardness in the world, then we can put it to good use! Without this theorem, one can imagine that all of crypto would break down once some specific assumption gets broken. But with it, we know that we are OK as long as one-way (or, for public-key encryption, trapdoor) permutations exist.

Another reason why I want to show it to you is that I think it is rather ingenuous and I hope you will be impressed.

Proof. It is sufficient to show that, if $p'_{PK}(\cdot)$ is a one-way permutation family of $S_{PK} \subseteq \{0,1\}^{\ell(PK)}$ then $p_{PK}(\cdot,\cdot)$ where the second argument is randomly chosen from $\{0,1\}^{\ell(PK)}$, and $p_{PK}(x,r) = p'_{PK}(x) \circ r$ is a one-way permutation with hardcore bit $b(x,r) = x.r$. It is clear that p is a OWP family. It is also clear that b gives one bit of information about its input.

Suppose for contradiction that $b(x,r)$ is not hardcore for p.

Then there exists a predictor $\{A_k\}$ and a polynomial $a(k)$ such that for infinitely many k's

$$p = \Pr[PK \leftarrow G(1^k); x \leftarrow S(PK); r \leftarrow \{0,1\}^{\ell(PK)}; b' = A_k(PK, p_{PK}(x,r) : b' = x.r] \geq 1/2 + 1/a(k)$$

Consider a value k for which the statement above is true. It is easy to see that for a polynomial fraction of all $PK \in G(1^k)$, call them the "good" PK's,

$$p_{good} = \Pr[x \leftarrow S(PK); r \leftarrow \{0,1\}^{\ell(PK)}; b' = A_k(PK, p_{PK}(x,r) : b' = x.r] \geq 1/2 + 1/2a(k)$$

Why? Let $g = \Pr[PK \leftarrow G(1^k) : PK \text{ is good}]$. Suppose $g < 1/a(k)$. Then

$$\begin{aligned}
p &= p_{good} * g + p_{bad} * (1-g) \\
&\leq 1 * g + (1/2 + 1/2a(k))(1-g) \\
&= 1/2 + 1/2a(k) + g(1/2 - 1/2a(k)) \\
&< 1/2 + 1/2a(k) + 1/2a(k) - 1/2a^2(k) \\
&< 1/2 + 1/a(k)
\end{aligned}$$

But we know that $p > 1/2 + 1/a(k)$, so this is a contradiction.

By the same token, for all good PK's, for a polynomial fraction of $x \in S(PK)$ (call these x's "good")

$$\Pr[r \leftarrow \{0,1\}^{\ell(PK)}; b' \leftarrow A_k(PK, p_{PK}(x,r)) : b' = x.r] \geq 1/2 + 1/4a(k)$$

Suppose that PK and x are good, and that we are given $y = p'_{PK}(x)$. Since this occurs with non-negligible probability, it is enough to show how to invert p_{PK} when we are restricted to this situation.

Let us show how to build an inverter R_k that inverts $p_{PK}(\cdot)$ using the predictor A_k.

Before we go any further, let us get some intuition. Imagine that, somehow, we already know the value $x.r^j$ for a whole bunch (say, $\ell(4a(k))^2$) of values $\{r^j\}$ [1]. How can we use this information and the predictor A_k to give us x? Well, if the r^j's are totally random, we just divide them into batches of $(4a(k))^2$ each. Use batch number i to compute bit x_i by running A_k on input $(PK, (y \circ (r \oplus e_i)))$, where e_i denotes a string of all but one 0's, with a 1 in position i. Significantly more than one half of the tests in batch i will give the right value for $x.(r^j \oplus e_i)$. Now, what we are after is actually $x.e_i$. How do we get that? Observing that $x.e_i = x.(r \oplus (r \oplus e_i)) = (x.r) \oplus (x.(r \oplus e_i))$, we will do it by taking the majority of $(x.r^j) \oplus (x.(r^j \oplus e_i))$.

How do we get the values $x.r^j$ for a bunch of totally random r^j's? The answer is: we won't. If our r^j's aren't totally random but are *pairwise independent*. We don't have to compute r^j's, but can just guess them correctly with good probability. This will also be sufficient, as it turns out, for the majority of the tests to give us the right answer for $x.(r^j \oplus e_i)$.

(Recall that random variables X and Y are pairwise independent if for any value y of the random variable Y, $\Pr[X|Y = y] = \Pr[X]$ and for any value x of the random variable X, $\Pr[Y|X = x] = \Pr[Y]$.)

Exactly how we do that:

To generate L pairwise independent r's with probability $1/L$ correctly guessed $x.r^j$,

- First, pick $u = \lceil \log L + 1 \rceil$ randoms s^l's, make a guess σ^l for the value $x.s^l$.
- For all $J \subseteq [u] = [1, ..., u]$ (there are $2^\ell \approx L$ such J's), let $r^J = \bigoplus_{l \in J} s^l$, so $x.r^J = \bigoplus_{l \in J} x.s^l$), and so our guess for $x.r^J$ is $\rho^J = \bigoplus_{l \in J} \sigma^l$).

It is easy to see that (1) the values r^J are pairwise independent, and (2) IF for all $1 \leq l \leq u$, σ^l is the right guess for $x.s^l$ (call this event A), THEN for all $J \subseteq [u]$, ρ^J is the right guess for $x.r^J$.

Now we must show that the majority of A_k's answers for the values $x.(r^J \oplus e_i)$ are correct with probability at least $1 - 1/2k$, i.e.

$$\Pr[\text{For more than one half of } J\text{'s, } A_k(y, (r^J \oplus e_i)) = x.(r^J \oplus e_i)] \geq 1 - 1/2k$$

This follows by using some inequalities from probability theory.

Chebyshev's inequality: if X is a random variable, and $\delta > 0$, then: $\Pr[|X - E(X)| \geq \delta] \leq Var(X)/\delta^2$.

[1] r superscript j, not r to the jth

We want to use this inequality to bound the probability that the majority of the answers of A_k are incorrect. If our random variable X is the number of correct guesses, then $E[X] = L(1/2 + 1/4a(k))$, and δ that we are interested in is $\delta = L\epsilon$, i.e., we want to bound the probability that X deviates from its expected value from more than δ. So this would give us a good bound if we knew $Var(X)$.

Let us estimate $Var(X)$. Let us show the following:

Claim: If $X_1, \ldots, X_n$ are pairwise independent, then $Var[\sum X_i] = \sum Var[X_i]$.

Proof of claim: $Var[\sum X_i] = E[(\sum X_i)^2] - E^2[\sum X_i]$ essentially by definition of variance.

$$\begin{aligned} E[(\sum X_i)^2] &= E[\sum_{i=1}^{n} X_i^2 + \sum_{i \neq j} X_i X_j] \\ &= \sum_{i=1}^{n} E[X_i^2] + \sum_{i \neq j} E[X_i X_j] \\ &= \sum_{i=1}^{n} E[X_i^2] + \sum_{i \neq j} E[X_i]E[X_j] \end{aligned}$$

where the last two derivations follow by linearity of expectation and pairwise independence, respectively.

On the other hand,

$$\begin{aligned} E^2[\sum X_i] &= E[\sum X_i]E[\sum X_i] \\ &= (\sum E[X_i])(\sum E[X_i]) \\ &= \sum_{i=1}^{n} E^2[X_i] + \sum_{i \neq j} E[X_i]E[X_j] \end{aligned}$$

Putting these together:

$$\begin{aligned} Var[X] &= \sum_{i=1}^{n} E[X_i^2] + \sum_{i \neq j} E[X_i]E[X_j] - \sum_{i=1}^{n} E^2[X_i] - \sum_{i \neq j} E[X_i]E[X_j] \\ &= \sum_{i=1}^{n} (E[X_i^2] - E^2[X_i]) + \sum_{i \neq j} (E[X_i]E[X_j] - E[X_i]E[X_j] \\ &= \sum_{i=1}^{n} Var(X_i) \end{aligned}$$

Putting the claim together with Chebyshev's inequality, we get the following corollary:

Corollary: If $X_1, \ldots, X_n$ are pairwise independent, with expectations μ_i and variances (σ_i^2), then for every $\delta > 0$, $\Pr[|\sum X_i - \sum \mu_i| \geq \delta] \leq \sum \sigma_i^2/\delta^2$.

Let us use the corollary. In our case, $\mu_i = 1/2 + 1/4a(k)$, σ_i^2 can be upper-bounded by $1/4$, $n = L$, $\delta = L/4a(k)$. So overall, if we set L such that $2k \leq L/16a^2(k)$,

$$\begin{aligned}
\Pr[\text{For less than one half of } J\text{'s, } A_k(y,(r^J \oplus e_i)) = x.(r^J \oplus e_i)] &\leq \sum \sigma_i^2/\delta^2 \\
&\leq \frac{L/4}{(L/4a(k))^2} \\
&\leq \frac{16a^2(k)}{L} \\
&\leq 1/2k
\end{aligned}$$

And therefore, we are done.

□