COM S 687 Introduction to Cryptography January 29, 2008

Lecture 3: One-Way Functions

Instructor: Rafael Pass *Scribe: Tom Roeder*

1 Adversaries

One-way functions are one of the most fundamental cryptographic primitives; we will return to them frequently in this course. Recall that last time we talked about information-theoretic security and Shannon's definition. This definition is natural and useful, and the One-Time Pad (OTP) scheme is provably secure under this definition, but the OTP scheme requires keys to be as long as messages. Worse, Shannon's Theorem shows this property to be fundamental: given a key space $\mathcal{K}$, a message space $\mathcal{M}$, and an encryption scheme secure under Shannon's definition, it must be the case that $|\mathcal{K}| \geq |\mathcal{M}|$. This isn't a problem with the definition itself, but rather with the notion of encryption, since there is otherwise an exponential time attack on the scheme: just try all possible encryptions.

To get around the requirements of Shannon's definition and still arrive at a useful encryption scheme, we will consider computationally bounded adversaries. Originally, we allowed adversaries to be arbitrary Turing Machines (TMs), potentially with a random tape. Now, we consider a restricted class of adversaries:

Definition 1 *A TM M is said to be* efficient *if there is a polynomial p such that the running time of M is bounded by p on all inputs and all values of random tape for M.*

Efficient TMs are also called *Probabilistic Polynomial Time* (PPT) TMs (sometimes written simply *PPT*). All of our cryptographic schemes will also be PPTs. We will also consider a somewhat stronger notion of adversaries by weakening the requirement that the TM must run a constant-sized program for all inputs.

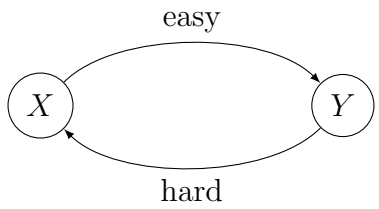
Definition 2 *A TM M is* non-uniform PPT *if it consists of a sequence of PPT machines $A = \{A_1, A_2, \ldots\}$ for which there is a polynomial d such that $|A_i| \leq d(i)$ and the running time of A_i on input in $\{0,1\}^i$ is also bounded by $d(i)$.*

Writing the sequence of A_i for all $i > 0$ is tedious, so, in practice, we write $A(x)$ to mean $A_{|x|}(x)$: the length of input x uniquely determines the machine to run. Note that non-uniform PPT machines not only run in polynomial time in their input but can also run a program that is polynomial in the length of their input. We will not, however,

allow protocols that run in non-uniform PPT time, since this class of computation does not seem to be feasible in practice.[1]

2 One-Way Functions

Last time, we mentioned candidates for hard problems, but hardness is not enough for cryptography; we need functions, like encryption, that are easy to compute but hard to invert. The general structure of a *one-way function* f with domain X and range Y is as follows



Physical examples of one-way functions include striking a match (easy to light, but hard to get a lightable match from a burnt one) and looking up numbers in the phone book (easy to look up a number, but hard to find a given number).

Consider multiplication as a mathematical candidate for being a one-way function: $f(x, y) = xy$ where $|x| = |y|$. Finding a product is easy, but factoring is (believed to be) hard. Formally, we can define different kinds of one-way functions.

Definition 3 *A function* $f : \{0,1\}^n \longrightarrow \{0,1\}^n$ *is said to be* worst-case one-way *if*

- f *can be computed in PPT*
- *there is no non-uniform PPT* A *such that for all integers* n *and all* x *in* $\{0,1\}^n$

$$\Pr[A(1^n, f(x)) \in f^{-1}(f(x))] = 1$$

Value 1^n (n written in unary notation) is given to A to avoid labeling $f(x) = |x|$ a worst-case one-way function; many such compression functions would otherwise trivially be one-way since inverting functions can only run in time polynomial in their input, and $f(x) = |x|$ with x chosen from $\{0,1\}^n$ gives output of size $\log n$. Thus, no inverting function has enough running time even to write down the answer.

[1]For example, some variants of the Halting Problem can be solved in non-uniform PPT but cannot be solved in PPT.

But worst-case one-way functions are not good enough for cryptography. In particular, an encryption function that was only a worst-case one-way function might be easy to invert on 99% of its keys; worst-case one-way functions only need to be hard on some input. Cryptographic uses require average-case hardness: it should be hard to invert the output of the function on a value randomly chosen from the domain, except with small probability. Before giving a stronger definition for one-way functions, we formalize the meaning of "small probability". The intuition is that a function is small if its output is smaller than the inverse of any polynomial.

Definition 4 *A function* $\epsilon : \mathbb{N} \longrightarrow \mathbb{R}$ *is said to be* negligible *if*

$$\forall c \in \mathbb{R}_{>0} \; . \; \exists n_0 \in \mathbb{N} \; . \; \forall n > n_0 \; . \; \epsilon(n) < \frac{1}{n^c}$$

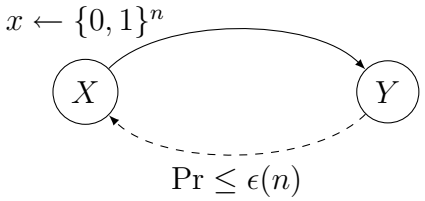
The definition of negligible is asymptotic; concrete security requires functions that are not just asymptotically smaller than polynomials, but that are very small for reasonable values. The asymptotic definitions, however, are cleaner for exposition and proofs, since, for instance, the sum and product of negligible functions is negligible, and even the product of a polynomial with a negligible function is negligible. Now we can define a more useful type of one-way function.

Definition 5 *A function* $f : \{0,1\}^n \longrightarrow \{0,1\}^n$ *is a* (strong) one-way function *if*

- f *can be computed in PPT*
- *For all non-uniform PPT* A*, there is a negligible* ϵ *such that for all integers* n*,*

$$\Pr[x \leftarrow \{0,1\}^n : A(1^n, f(x)) \in f^{-1}(f(x))] \leq \epsilon(n)$$

Now the picture looks something like the following



We now have a definition of one-way functions as well as a candidate, but multiplication is not a strong one-way function, since 3/4 of the time, one of x and y in $f(x, y) = xy$ is even, and even numbers are easy to factor. We can formalize our intuition about what is hard in multiplication:

Definition 6 *A function* $f : \{0,1\}^n \longrightarrow \{0,1\}^n$ *is a* weak one-way function *if*

- f *can be computed in PPT*
- *there is a polynomial* $q(n)$ *such that for all non-uniform PPT* A *and all integers* n*,*

$$\Pr[x \leftarrow \{0,1\}^n : A(1^n, f(x)) \in f^{-1}(f(x))] < 1 - \frac{1}{q(n)}$$

A weak one-way function is a one-way function where the probability of inverting the output of a randomly chosen value is on the order of $1 - \frac{1}{n}$, which is to say that there is a noticeable fraction of the time that the adversary cannot invert it. Strong or weak one-way functions are said to satisfy strong or weak *one-wayness*, respectively.

It is conjectured that multiplication is a weak one-way function: in particular, the product of two primes is believed to be hard to invert. One way to find a strong one-way function would be to construct it from a given weak one-way function. A potential construction is to concatenate the output of many invocations of a weak one-way function; intuitively, an adversary must then invert each of the outputs and there is likely to be one output that is hard to invert.

Formally, define a function $g(x_1, x_2, \ldots, x_m) = f(x_1)||f(x_2)||\cdots||f(x_m)$, where $||$ represents concatenation. If we take $m = nq(n)$, then an appealing—but wrong—argument for the strong one-wayness of g would be to note that the probability of the adversary's success in inverting any given output $f(x_i)$ is $1 - \frac{1}{q(n)}$, so the probability of inverting all of them is $(1 - \frac{1}{q(n)})^{nq(n)} \approx \frac{1}{e^n}$, which is negligible.

Unfortunately, this argument relies on the independence of the adversary's actions: it effectively assumes that each output $f(x_i)$ is tried independently in turn. The formal proof that g is a strong one-way function requires a result called the Hardness Amplification Lemma (to be proved next time).

Theorem 1 *If there exists a weak one-way function, then there exists a strong one-way function.*

Lemma 2 *Let* f *be a weak one-way function, and let* $q(n)$ *be the polynomial in the definition of weak one-way functions. Then*

$$g(x_1, x_2, \ldots, x_m) = f(x_1)||f(x_2)||\cdots||f(x_m)$$

is a strong one-way function with $m = 2nq(n)$*.*

The proof of this result will proceed along normal lines: we assume that there is an adversary A that can violate strong one-wayness of g and construct a new adversary A' that can violate weak one-wayness of f.

The importance of going from weak one-wayness to strong one-wayness is in making the weakest possible assumptions; since there are no provably one-way functions, it is better only to assume the existence of a weak one-way function rather than a strong one-way function.