# Examples of reductions for one-way functions

Noah Stephens-Davidowitz

May 29, 2023

These are supplemental notes in which I provide some examples of security proofs via a reduction. They should (hopefully) be helpful in showing you how to do the homework and in filling in some of the details that I do not have the time to cover in lectures.

The basic idea of a proof by reduction is to show that, if we can solve computational problem $A$ efficiently, then we can solve computational problem $B$ efficiently. We prove this by showing how to construct an algorithm that solves problem $B$ efficiently, *provided that it is given access to a subroutine that solves problem $A$*. Such an algorithm is called a *reduction* from $B$ to $A$. As cryptographers, we typically turn this on its head and take the contrapositive. In particular, if we have a reduction from $B$ to $A$, and we believe that $B$ is hard (i.e., that $B$ cannot be solved efficiently), then we must also believe that $A$ is hard.

(Note that it is *very* easy to get confused about directions here. I.e., it is very easy to accidentally show a reduction from $A$ to $B$, when you meant to show a reduction from $B$ to $A$. It is always good to step back and ask what you're trying to prove. E.g., if you're trying to prove that $A$ is hard, then you want to show that an efficient algorithm for $A$ would imply something absurd: e.g., an efficient algorithm for $B$.)

Such proofs are the bread and butter of cryptography, and we use them *a lot*. Reductions in cryptography also tend to be quite subtle for a number of reasons. Perhaps most importantly, in cryptography the problems ($A$ and $B$) that we study are almost always *average-case* problems, and we are usually interested in whether there exist efficient algorithms that solve $A$ and $B$ with non-negligible probability or with non-negligible advantage.

E.g., consider the example of one-way functions. Suppose that we have some construction of a one-way function $g$, and we would like to show that it is secure (i.e., hard to invert) under the assumption that some other one-way function $f$ is secure. To do this, we first take the contrapositive: we show that if $g$ were *insecure*, then $f$ would necessarily be insecure. To do so, we want to show a reduction from the problem of breaking $f$ to the problem of breaking $g$. In other words, we assume that we have access to a PPT adversary $\mathcal{A}$ that inverts $g$ with non-negligible probability. That is,

$$\Pr_{x\sim\{0,1\}^n}[x' \leftarrow \mathcal{A}(1^n, g(x)) \ : \ g(x) = g(x')] \geq \varepsilon(n) \ , \tag{1}$$

for some *non-negligible* $\varepsilon(n)$. We then show that we could use such an $\mathcal{A}$ to construct a new adversary $\mathcal{A}'$ that inverts $f$ with non-negligible probability, i.e., such that

$$\Pr_{y\sim\{0,1\}^m}[y' \leftarrow \mathcal{A}'(1^m, f(y)) \ : \ f(y) = f(y')] \geq \varepsilon'(m) \ ,$$

for some non-negligible $\varepsilon'$. It is crucial to make sure that your proof only uses Eq. (1) and does not assume anything else about $\mathcal{A}$. For example, you *may not* assume that $\mathcal{A}$ *always* finds an inverse

to $g(x)$ for any input $x$, or that $\mathcal{A}$ has non-negligible probability of inverting $g(x)$ when $x \sim D$ is sampled from some other distribution $D$, as opposed to the uniform distribution $x \sim \{0,1\}^n$. This is crucial, as some of the below examples illustrate.

It is *very* often the case that a statement in cryptography looks quite obvious but is quite difficult to prove via reduction. This simply comes with the territory. Indeed, it is also very often the case that a statement in cryptography looks quite obvious but is actually false! Even experts get this stuff wrong all the time! This is why our proofs must be so careful.

Below, I do some examples. For some, I will include a very pedantic version and a less pedantic version of the same proof—so that if you're confused by the less pedantic version, you can check the details in the pedantic version. (The less pedantic proofs are what we will typically use—and what I expect from you in your homework—but it's important that we are always sure that such proofs can be trivially expanded to a fully formal pedantic proof without any subtle issues arising.)

**Claim 1.** *Let $g(x_1|x_2) := f_1(x_1)|f_2(x_2)$ (where here we use $|$ to represent string concatenation and $x_1$ and $x_2$ have the same length), where $f_1$ and $f_2$ are both efficiently computable functions. Then, $g$ is a secure one-way function if* either *$f_1$ is a secure one-way function* or *$f_2$ is a secure one-way function.*

*Proof.*

**Standard proof.** We assume for contradiction that $g$ is not secure, i.e., that there exists a PPT adversary $\mathcal{A}$ such that

$$\Pr_{x \sim \{0,1\}^n}[x' \leftarrow \mathcal{A}(1^n, g(x)) \ : \ g(x) = g(x')] \geq \varepsilon(n) \ ,$$

for some *non-negligible* $\varepsilon(n)$. Then, we construct two PPT adversaries $\mathcal{B}_1$ and $\mathcal{B}_2$, one that breaks $f_1$ and one that breaks $f_2$, contradicting the assumption that at least one of them is secure.

Our adversary $\mathcal{B}_1$ takes as input $y_1 := f_1(x_1)$, where $x_1 \sim \{0,1\}^m$; samples $x_2 \sim \{0,1\}^m$; sets $y_2 := f_2(x_2)$; runs $x_1'|x_2' \leftarrow \mathcal{A}(1^{2m}, y_1|y_2)$; and outputs $x_1'$. Clearly, $\mathcal{B}_1$ runs in polynomial time, and the input to $\mathcal{A}$ in the reduction is identical to its input in the security game against $g$. Therefore,

$$\Pr[f_1(x_1') = f_1(x_1)] \geq \Pr[g(x_1'|x_2') = g(x_1|x_2)] \geq \varepsilon(2m) \ ,$$

which is non-negligible. This implies that $f_1$ cannot be a one-way function if $g$ is not a one-way function.

A similar proof shows that $f_2$ cannot be a one-way function if $g$ is not a one-way function, and we are done.

**Pedantic version of the proof.** We prove this via reduction. Specifically, we assume that there exists a PPT adversary $\mathcal{A}$ such that

$$\Pr_{x \sim \{0,1\}^n}[x' \leftarrow \mathcal{A}(1^n, g(x)) \ : \ g(x) = g(x')] \geq \varepsilon(n) \ ,$$

for some *non-negligible* $\varepsilon(n)$. Then, we construct two PPT adversaries $\mathcal{B}_1$ and $\mathcal{B}_2$, one that breaks $f_1$ with non-negligible probability and one that breaks $f_2$ with non-negligible probability. This would contradict the assumption that either $f_1$ or $f_2$ is one way, as needed.

We construct $\mathcal{B}_1$ as follows. $\mathcal{B}_1$ takes as input $1^m$ and $y_1 := f_1(x_1)$, where $x_1 \sim \{0,1\}^m$. (I.e., $\mathcal{B}_1$ is playing the one-way function game against $f_1$. To be clear, $x_1$ is not part of the input, and is unknown to $\mathcal{B}_1$.) It then samples $x_2 \sim \{0,1\}^m$ itself and sets $y_2 := f_2(x_2)$. Finally, it calls $\mathcal{A}$ on input $1^{2m}$ and $y_1|y_2$, receiving as output $x_1'|x_2'$, and it outputs $x_1'$.

To see that this reduction works, first notice that it runs in polynomial time, since it simply samples $m$ uniformly random bits; computes $f_2$, which is efficiently computable by assumption; and concatenates some strings together. Next, we claim that

$$\Pr_{x_1 \sim \{0,1\}^m}[x_1' \leftarrow \mathcal{B}_1(1^m, f_1(x_1)) \ : \ f_1(x_1') = f_1(x_1)] \geq \varepsilon(2m) \ ,$$

which is non-negligible. To see this, first, notice that if $x_1 \sim \{0,1\}^m$ and $y_1 := f_1(x_1)$, then the input to $\mathcal{A}$ is distributed exactly as $g(x)$ where $x \sim \{0,1\}^{2m}$. Therefore, by assumption

$$\Pr[g(x_1'|x_2') = g(x_1|x_2)] \geq \varepsilon(2m) \ .$$

Finally, we notice that whenever $g(x_1'|x_2') = g(x_1|x_2)$, we must have $f_1(x_1) = f_1(x_1')$. Therefore, $\mathcal{B}_1$ inverts $f_1$ (with uniformly random input) with non-negligible probability, contradicting the assumption that $f_1$ is a one-way function.

The construction of $\mathcal{B}_2$ is identical, except with the roles of $f_1$ and $f_2$ reversed, and the proof that it works is identical. (Even in very pedantic proofs, we do not waste ink by writing the same argument twice.) □

**Claim 2.** *If $f$ is a one-way function, then $g(x) := f(x[1, \sqrt{|x|}])$ is a one-way function. I.e., $g$ depends only on the first $\sqrt{n}$ bits of its input, where $n := |x|$ is the length of the input to $g$.*

*Proof.*

**Standard proof.** We assume for contradiction that $g$ is not secure, i.e., that there exists a PPT adversary $\mathcal{A}$ such that

$$\Pr_{x \sim \{0,1\}^n}[x' \leftarrow \mathcal{A}(g(x)) \ : \ g(x) = g(x')] \geq \varepsilon(n) \ ,$$

for some *non-negligible* $\varepsilon(n)$. Then, we construct a PPT adversary $\mathcal{B}$ that breaks $f$.

Our adversary $\mathcal{B}$ takes as input $1^m$ and $y := f(x)$, where $x \sim \{0,1\}^m$; sets $x' \leftarrow \mathcal{A}(1^{m^2}, y)$; and outputs $x'[1, \sqrt{|x'|}]$. Clearly, $\mathcal{B}$ runs in polynomial time, and the input to $\mathcal{A}$ in the reduction is identical to its input in the security game against $g$. Furthermore, we have $f(x'[1, \sqrt{|x'|}]) = y$ if and only if $g(x') = y$. Therefore, $\Pr[f(x'[1, \sqrt{|x'|}]) = y] \geq \varepsilon(m^2)$, which is non-negligible, contradicting the assumption that $f$ is a one-way function.

**Pedantic version of the proof.** We prove this via reduction. Specifically, we assume that there exists a PPT adversary $\mathcal{A}$ such that

$$\Pr_{x \sim \{0,1\}^n}[x' \leftarrow \mathcal{A}(1^n, g(x)) \ : \ g(x) = g(x')] \geq \varepsilon(n) \ ,$$

for some *non-negligible* $\varepsilon(n)$ and show that we can use this to construct a PPT adversary $\mathcal{B}$ that breaks $f$. Since $f$ is secure by assumption, this is a contradiction.

We construct $\mathcal{B}$ as follows. $\mathcal{B}$ takes as input $1^m$ and $y := f(x)$, where $x \sim \{0,1\}^m$. It simply calls $\mathcal{A}$ on input $1^{m^2}$ and $y$, receiving as output $x' \in \{0,1\}^\ell$. Finally, it outputs $x'[1,\sqrt{\ell}]$ (i.e., it outputs the first $\sqrt{\ell}$ bits of $x'$—notice that it is not necessarily the case that $\ell = m^2$).

To see that this reduction works, first notice that it runs in polynomial time. Next, we claim that

$$\Pr_{x\sim\{0,1\}^m}[x' \leftarrow \mathcal{B}(1^m, f(x)) \ : \ f(x'[1,\sqrt{\ell}]) = f(x)] \geq \varepsilon(m^2) \ ,$$

which is non-negligible.[1] To see this, first, notice that if $x \sim \{0,1\}^m$ and $y := f(x)$, then the input to $\mathcal{A}$ is distributed exactly as $g(x^*)$ where $x^* := x|x''$ with $x'' \sim \{0,1\}^{m^2-m}$. I.e., $x^*$ is distributed identically to a uniformly random $m^2$-bit string. Therefore, by assumption

$$\Pr[g(x') = g(x^*)] \geq \varepsilon(m^2) \ .$$

Notice that if $\varepsilon$ is a non-negligible function, then $\varepsilon'(m) := \varepsilon(m^2)$ is also a non-negligible function (ignoring the annoying issues discussed in the footnote). Finally, we notice that whenever $g(x') = g(x^*)$, we must have $f(x) = f(x'[1,\sqrt{\ell}])$ by definition. Therefore, $\mathcal{B}$ inverts $f$ (with uniformly random input) with non-negligible probability, contradicting the assumption that $f$ is a one-way function. □

**Claim 3.** *Let $h$ be a one-way function. Then,*

$$\varepsilon(n) := \Pr_{x\sim\{0,1\}^n}[h(x) = 0]$$

*is negligible.*

*Proof.* Consider the adversary $\mathcal{A}$ that on input $(1^n, y)$ simply samples a uniformly random string $x' \sim \{0,1\}^n$ and returns $x'$ (without bothering to look at $y$). Notice that

$$\varepsilon'(n) := \Pr_{x\sim\{0,1\}^n}[x' \leftarrow \mathcal{A}(1^n, h(x)) \ : \ h(x) = h(x')] = \Pr_{x,x'\sim\{0,1\}^n}[h(x) = h(x')] \geq \Pr_{x\sim\{0,1\}^n}[h(x) = 0]^2 = \varepsilon(n)^2 \ .$$

By assumption, this probability is negligible. Finally, we recall from Homework 1 that if $\varepsilon'(n)$ is negligible then so is $\varepsilon(n) = (\varepsilon'(n))^{1/2}$. □

The following example highlights the fact that a one-way function is only hard to invert on uniformly random input, not necessarily on all input.

**Claim 4.** *If one-way functions exist, then there exists a one-way function $f$ such that $g(x) := f(0^{|x|}|x)$ is* not *a one-way function.*

*Proof.*

[1] There's actually an annoying subtlety here that we are ignoring, even in this super pedantic version of the proof. The issue has to do with the fact that we did not really specify how the function $g$ behaves when its input has length that is not a perfect square. E.g., suppose we specify that $g$ depends on the first $\lfloor\sqrt{n}\rfloor$ bits in this case. And, suppose that our adversary $\mathcal{A}$ is super sneaky, in the sense that it only breaks $g$ when its input is $(1^n, y)$ for some $n$ that is *not* a perfect square. Then, the above proof would not work, since the above proof only calls the adversary with security parameter equal to a perfect square. (We do formally need to worry about such things. Adversaries are adversarial!) Indeed, it is not necessarily the case that $\varepsilon'(m) := \varepsilon(m^2)$ is non-negligible if $\varepsilon(n)$ is non-negligible, because of annoying functions like

$$\varepsilon(n) := \begin{cases} 2^{-n} & n \text{ is a perfect square} \\ 1/n & \text{otherwise.} \end{cases}$$

We ignore issues like this unless they are central to the problem that we are solving, knowing that we could resolve them if we had to. (In this case, one can fix this, e.g., by choosing a uniformly random integer $m'$ between $m^2$ and $(m+1)^2$ and calling $\mathcal{A}$ on input $(1^{m'}, y)$.

**Pedantic version of the proof.** Let $h$ be any one-way function (which exists by assumption). Then, define

$$f(x_1|x_2) := \begin{cases} 0 & x_1 = 0^{|x_1|} \\ h(x_1|x_2) & \text{otherwise,} \end{cases}$$

where $|x_1| = |x_2|$. Notice that $g(x) := f(0^{|x|}|x)$ is clearly not a one-way function, since $g(x) = 0$ for all $x$. In particular, an adversary can invert $g$ by literally outputting any bit string at all.

So, it remains to show that $f$ is in fact a one-way function (assuming that $h$ is). First, notice that $f$ is in fact efficiently computable. Now, suppose for the sake of contradiction that $f$ is not one way, i.e., that there exists a PPT adversary $\mathcal{A}$ such that

$$\Pr_{x \sim \{0,1\}^n}[x' \leftarrow \mathcal{A}(1^n, f(x)) \ : \ f(x) = f(x')] \geq \varepsilon(n) \ ,$$

for some non-negligible $\varepsilon(n)$. We claim that

$$\Pr_{x \sim \{0,1\}^n}[x' \leftarrow \mathcal{A}(1^n, h(x)) \ : \ h(x) = h(x')] \geq \varepsilon(n) - \varepsilon'(n)$$

for some negligible $\varepsilon'$. (Recall that we proved in Homework 1 that $\varepsilon(n) - \varepsilon'(n)$ is non-negligible if $\varepsilon$ is non-negligible and $\varepsilon'$ is negligible.) In other words, we claim that $\mathcal{A}$ can itself be viewed as an adversary against $h$ with non-negligible advantage. This would contradict the assumption that $h$ is a one-way function. So, we must have that $f$ is a one-way function, as needed.

To see this, let $E_x$ be the event that the first $|x|/2$ bits of $x$ are all equal to zero and $\neg E_x$ be the complement of this event. Notice that

$$\begin{aligned} \Pr_{x \sim \{0,1\}^n}[x' \leftarrow \mathcal{A}(1^n, h(x)) \ : \ h(x) = h(x')] &\geq \Pr_{x \sim \{0,1\}^n}[x' \leftarrow \mathcal{A}(1^n, h(x)) \ : \ h(x) = h(x') \text{ and } \neg E_x \text{ and } h(x) \neq 0] \\ &= \Pr_{x \sim \{0,1\}^n}[x' \leftarrow \mathcal{A}(1^n, f(x)) \ : \ f(x) = f(x') \text{ and } \neg E_x \text{ and } h(x) \neq 0] \\ &\geq \Pr_{x \sim \{0,1\}^n}[x' \leftarrow \mathcal{A}(1^n, f(x)) \ : \ f(x) = f(x')] - \Pr[E_x] - \Pr[h(x) = 0] \ , \end{aligned}$$

where in the last step we have used the union bound, which in particular says that $\Pr[A \text{ and } B \text{ and } C] \geq \Pr[A] - \Pr[\neg B] - \Pr[\neg C]$.

It remains to show that $\Pr[E_x]$ and $\Pr[h(x)]$ are both negligible. Indeed, we trivially have $\Pr[E_x] = 2^{-n/2}$, which is clearly negligible. The fact that $\Pr[h(x) = 0]$ is negligible is exactly Claim 3.

**Standard proof.** Here, we use a slightly different construction that makes the proof more elegant (and avoids needing to use Claim 3). Let $h$ be any one-way function, and let

$$f(x_1|x_2) := \begin{cases} 0 & x_1 = 0^{|x_1|} \\ 1|h(x_1|x_2) & \text{otherwise,} \end{cases}$$

where $|x_1| = |x_2|$. (Notice that we have appended one to the output when $x_1 \neq 0^{|x_1|}$, so that now we simply cannot have $f(x_1|x_2) = 0$ unless $x_1 = 0^{|x_1|}$. This is much more convenient. Incidentally, this simple technique is sometimes called "range separation," since we are explicitly making sure that the range of $f$ in the two cases is distinct.)

Notice that $g(x) := f(0^{|x|}|x)$ is clearly not a one-way function, since $g(x) = 0$ for all $x$. (Indeed, any adversary that outputs any bit string $x' \in \{0,1\}^*$ at all will break this $g$, since $g(x) = g(x')$ for all $x, x' \in \{0,1\}^*$, which is rather silly :P.) So, we only need to prove that $f$ is a one-way function (assuming that $h$ is a one-way function).

Suppose for the sake of contradiction that $f$ is not one way, i.e., that there exists a PPT adversary $\mathcal{A}$ such that

$$\Pr_{x \sim \{0,1\}^n}[x' \leftarrow \mathcal{A}(1^n, f(x)) \ : \ f(x) = f(x')] \geq \varepsilon(n) \ ,$$

for some non-negligible $\varepsilon(n)$.

We construct an adversary $\mathcal{A}'$ for $h$ as follows. $\mathcal{A}'$ takes as input $1^n$ and $y := h(x)$ for $x \sim \{0,1\}^n$. It simply sets $x' \leftarrow \mathcal{A}(1^n, 1|y)$ and outputs $x'$.

We claim that

$$\Pr_{x \sim \{0,1\}^n}[x' \leftarrow \mathcal{A}'(1^n, h(x)) \ : \ h(x) = h(x')] \geq \varepsilon(n) - 2^{-n/2} \ ,$$

which is non-negligible. This would contradict our assumption that $h$ is one-way, as needed.

To see this, notice that

$$\begin{aligned}\Pr_{x \sim \{0,1\}^n}[x' \leftarrow \mathcal{A}'(1^n, h(x)) \ : \ h(x) = h(x')] &\geq \Pr_{x \sim \{0,1\}^n}[x' \leftarrow \mathcal{A}(1^n, f(x)) \ : \ f(x) = f(x') \text{ and } x[1, n/2] \neq 0^{n/2}] \\ &\geq \varepsilon(n) - 2^{-n/2} \ ,\end{aligned}$$

as needed, where we have used the fact that $h(x) = h(x')$ whenever $f(x) = f(x')$ and $x[1, n/2] \neq 0^{n/2}$. □