Scribe 2: Game-based proofs

**Disclaimer:** These notes have not been subjected to the usual scrutiny reserved for formal publications. They may be distributed outside this class only with the permission of the Instructor.

A problem. Let $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a good blockcipher, meaning it is a good PRF. Let $F : \{0,1\}^k \times \{0,1\}^{n-1} \to \{0,1\}^{2n}$ be defined by $F_K(x) = E_K(1\|x) \oplus E_K(0\|x)$. We want to prove that $F$ is also a good PRF.

Some comments. Before we get into the proof, here are some remarks. In this kind of problems, we’ll have to start from some game $G_0$, and then hop through a sequence of games $G_1 \to \cdots \to G_t$. The starting game $G_0$ and the terminal game $G_t$ will be the games in the definition in which we want to distinguish. For example, here since we are attacking the PRF security of $F$, and $G_0$ is the real game $\text{Real}_F^A$ and $G_t$ is the random game $\text{Rand}_F^A$. Our tasks are: (1) find the middle games $G_1, \ldots, G_{t-1}$, and (2) bound the gaps between the games.

Now, recall that $F$ is built on top of $E$, so one should write game $G_0$ in terms of $E$, not $F$. We then try to replace $E_K$ by a random function $f \leftarrow \text{Func}(n)$, where $\text{Func}$ is the set of all functions $g : \{0,1\}^n \to \{0,1\}^n$, resulting in game $G_1$. To account for the gap between $G_0$ and $G_1$, we’ll build an adversary $B$ attacking $E$, such that $G_0$ becomes $\text{Real}_E^B$, and $G_1$ becomes $\text{Rand}_E^B$. Thus $\Pr[G_0 \Rightarrow \text{true}] - \Pr[G_1 \Rightarrow \text{true}] \leq \text{Adv}_{E}^{\text{prf}}(B)$.

Next, we try to replace $f(\cdot)$ by the function $\$ (\cdot)$, namely the function that always returns a fresh $n$-bit random string on each input. Recall that the key difference between $f$ and $\$ is that the former remembers the past answers—if you repeat a prior query, you’ll get the same prior answer—whereas the latter remembers nothing. The change from $f$ to $\$ may cost us a few games. Occasionally, the change might be problematic, as we shall see in our problem. In that case, we need to re-examine the adversary $A$; often we can assume that $A$ belongs to a certain restricted class without loss of generality, but this restriction will smoothen the change from $f$ to $\$$. In our problem, we will assume that $A$ never repeats a prior query: the adversary can store the queries/answers it receives, and in both games $\text{Real}_E^A$ and $\text{Rand}_E^A$, repeating a prior query will result in the same prior answer, which the adversary can retrieve from its storage without querying. Once we are in a game using $\$ (\cdot)$, it’ll be essentially the random game $\text{Rand}_E^B$; one might need some equivalent transitions to make this explicit.

In each game, we’ll run $A$ to produce a bit $b'$—we often write $b' \leftarrow A^{\text{ENC}}$, meaning that the adversary $A$ interacts with the oracle $\text{ENC}$ and then outputs a bit $b'$. The game then returns $b' = 1$). Why returns $(b' = 1)$? For example, if $G_0 \Rightarrow \text{true}$, it means that $A$ outputs $b' = 1$ in the real game, in other words, $\text{Real}_F^A \Rightarrow 1$. Likewise, if $G_t \Rightarrow \text{true}$, it means that $A$ outputs $b' = 1$ in the random game, in other words, $\text{Rand}_F^A \Rightarrow 1$. Hence

$$\Pr[G_0 \Rightarrow \text{true}] - \Pr[G_t \Rightarrow \text{true}] = \Pr[\text{Real}_F^A \Rightarrow 1] - \Pr[\text{Rand}_F^A \Rightarrow 1] = \text{Adv}_{F}^{\text{prf}}(A) .$$

The proof. Let $A$ be an efficient adversary attacking $F$. Consider the following games $G_0$–$G_4$ in Figure 3.1. Game $G_0$ corresponds to the real game $\text{Real}_F^A$, and game $G_4$ corresponds to the random game $\text{Rand}_F^A$.

We now describe the game chain. Game $G_1$ is identical to game $G_0$, except that calls to $E_K$ are replaced by

---

If we blindly impose this restriction, it may actually affect the complexity of $A$, as it takes some additional cost (in both space and time) to store/retrieve the queries/answers. Thus the placement of the assumption needs to be handled with care.
Lecture 2: Game-based proofs

Figure 2.1: Games $G_0$–$G_4$. Here $\text{Func}(n - 1, 2n)$ is the set of all functions $h : \{0,1\}^{n-1} \rightarrow \{0,1\}^{2n}$.

<table>
<thead>
<tr>
<th>Procedure Enc($x$)</th>
<th>Game $G_2$</th>
<th>Game $G_3$</th>
<th>Game $G_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$y \leftarrow \text{Func}(n)$; $b' \leftarrow A^{\text{Enc}}$</td>
<td>$y_1 \leftarrow {0,1}^n$; $y_2 \leftarrow {0,1}^n$</td>
<td>$g \leftarrow \text{Func}(n - 1, 2n)$; $b' \leftarrow A^{\text{Enc}}$</td>
<td>$y \leftarrow {0,1}^{2n}$; Return $y$</td>
</tr>
<tr>
<td>Return ($b' = 1$)</td>
<td>Return $y_1</td>
<td></td>
<td>y_2$</td>
</tr>
</tbody>
</table>

Figure 2.2: Constructed adversary $B$ attacking $E$ from a given adversary $A$ attacking $F$.

the corresponding call to a truly random function $f \leftarrow \text{Func}(n)$. To bound the gap between $G_0$ and $G_1$, we construct an adversary $B$ as in Figure 3.2. Specifically, adversary $B$ runs $A$. Each time the latter queries $x$, the former queries $1||x$ and $0||x$ to its own oracle $F_{\text{N}}$ to get answers $y_1$ and $y_2$ respectively, and then gives $y_1||y_2$ back to $A$. Finally, when $A$ terminates and outputs a bit $b'$, adversary $B$ also outputs the same bit. Note that the adversary $B$ is about as efficient as $A$. Moreover, game $\text{Real}_E^B$ corresponds to game $G_0$, and game $\text{Rand}_E^B$ corresponds to game $G_1$, and thus

$$\Pr[G_0 \Rightarrow \text{true}] - \Pr[G_1 \Rightarrow \text{true}] = \Pr[\text{Real}_E^B] - \Pr[\text{Rand}_E^B] = \text{Adv}_E^{\text{prf}}(B).$$

We now bound the gap between $G_1$ and $G_4$. We claim that the gap between these two games is 0, for all (even computationally unbounded) adversaries $A$. We will now justify this claim. Since we consider computationally unbounded adversaries, without loss of generality, assume that the adversary $A$ does not repeat a prior query. Note that if $A$ makes $q$ distinct queries to $\text{Enc}$, in game $G_1$, it will result in $2q$ distinct queries to $f$. Since $f \leftarrow \text{Func}(n)$, the $2q$ answers will be independent and uniformly random. Thus we can rewrite game $G_1$ as in game $G_2$, in which for each query $\text{Enc}(x)$, instead of calling $y_1 \leftarrow f(1||x)$ and $y_2 \leftarrow f(0||x)$, we sample $y_1$ and $y_2$ at random. As game $G_2$ is equivalent to game $G_1$,

$$\Pr[G_2 \Rightarrow \text{true}] = \Pr[G_1 \Rightarrow \text{true}] .$$

Game $G_3$ is a further simplification of $G_2$. Instead of sampling $y_1, y_2 \leftarrow \{0,1\}^n$ and returning $y \leftarrow y_1||y_2$, we’ll pick $y \leftarrow \{0,1\}^{2n}$. Hence

$$\Pr[G_3 \Rightarrow \text{true}] = \Pr[G_2 \Rightarrow \text{true}] .$$

In game $G_4$, we explicitly sample a truly random function $g : \{0,1\}^{n-1} \rightarrow \{0,1\}^{2n}$, and for each query $\text{Enc}(x)$, we return $g(x)$. Since $A$ does not repeat prior queries, making $q$ distinct queries to $g(x)$ will result in $q$ independent, uniformly random answers. Hence games $G_3$ and $G_4$ are equivalent, and thus

$$\Pr[G_4 \Rightarrow \text{true}] = \Pr[G_3 \Rightarrow \text{true}] .$$

Hence $\Pr[G_1 \Rightarrow \text{true}] = \Pr[G_4 \Rightarrow \text{true}]$ as claimed. Summing up,

$$\text{Adv}_E^{\text{prf}}(A) = \Pr[G_0 \Rightarrow \text{true}] - \Pr[G_4 \Rightarrow \text{true}]$$

$$= (\Pr[G_0 \Rightarrow \text{true}] - \Pr[G_1 \Rightarrow \text{true}]) + (\Pr[G_1 \Rightarrow \text{true}] - \Pr[G_4 \Rightarrow \text{true}]) = \text{Adv}_E^{\text{prf}}(B) .$$

Hence if $E$ is a secure PRF then $F$ is also a secure PRF.