CSA E0 235: Cryptography 4 Sep 2019

Lecture 8

Instructor: Arpita Patra *Submitted by: Eklavya Sharma (16644)*

Contents

1 CPA-secure scheme using PRF

Recall from the previous lecture the scheme which used a truly random function (TRF) f as key. To encrypt a message m, a random string r is chosen and the ciphertext is $(r, f(r) \oplus m)$. The intuitive argument for its security is that since f is a TRF, $f(r)$ is completely random every time (if the same value of r is never used twice) and hence the scheme behaves like the one-time-pad with the pad stretching across multiple messages.

However, since the number of TRFs is very large (2^{n2^n}), there is no succinct representation for them. To overcome this problem, we will instead use Pseudorandom functions (PRFs).

Definition 1.1 (SKE using PRF)**.** *Let F be a keyed length-preserving PRF family. We will construct the scheme Π as follows:*

- ***Gen**: $k \in_R \mathbb{B}^n$ ($\mathbb{B} = \{0,1\}$)*
- ***Enc**: $e_k(m) = (r, F_k(r) \oplus m)$*
- ***Dec**: $d_k((r,c)) = F_k(r) \oplus c$*

Theorem 1. *F is PRF $\implies$ Π is CPA-secure.*

Proof. The contrapositive is: Π is not CPA-secure $\implies$ F is not a PRF. To prove this, we will use a successful adversary A for scheme Π to construct a distinguisher D for function family F.

Let Func_n refer to the set of all functions with n-bit input and output.

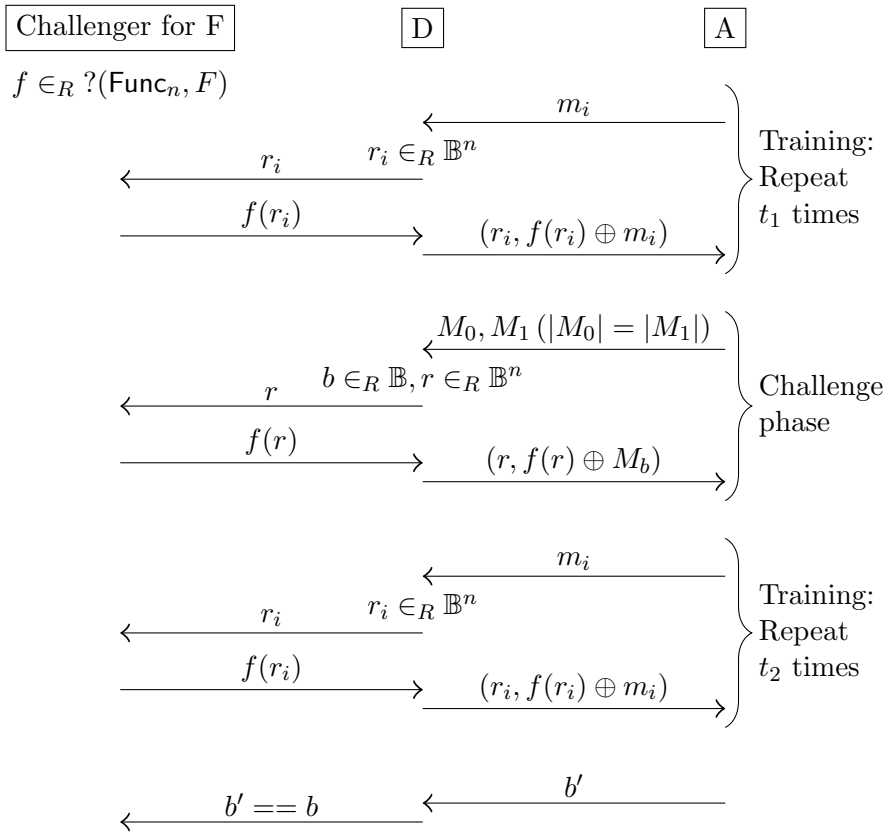

Figure 1: Constructing a distinguisher D for F which uses adversary A

Claim 2. *If $f \in_R F$, then A observes Π.*

Lemma 3. *For some polynomial $p(n)$,*

$$\Pr_{f \in_R F}[D^f(1^n) = 1] \geq \frac{1}{2} + \frac{1}{p(n)}$$

Proof.

$$\begin{aligned}
&\Pr_{f \in_R F}[D^f(1^n) = 1] \\
&= \Pr_{f \in_R F}[b' = b] \\
&= \Pr\left[\mathrm{PrivK}^{\mathrm{cpa}}_{A,\Pi}(n) = 1\right] && \text{(by claim 2)} \\
&\geq \frac{1}{2} + \frac{1}{p(n)} && \text{(because } A \text{ is a successful adversary for } \Pi\text{)}
\end{aligned}$$

□

Let $\widetilde{\Pi}$ be the encryption scheme observed by A when $f \in_R \mathsf{Func}_n$. Let Rep be the event that some r_i equals r.

Consider the scenario where $r_i = r$. A can recover $f(r)$ from the ciphertext by XOR-ing with m_i. Then in the challenge phase, it can XOR the ciphertext with $f(r)$ to get M_b. Therefore, A can break $\widetilde{\Pi}$ if Rep happens.

However, we will prove that Rep will happen rarely and the adversary can't break $\widetilde{\Pi}$ if Rep doesn't happen.

Lemma 4.

$$\Pr\left[\mathrm{PrivK}^{cpa}_{A,\widetilde{\Pi}}(n) = 1 \;\middle|\; \overline{\mathsf{Rep}}\right] = \frac{1}{2}$$

Proof. If Rep doesn't happen, then $f(r)$ is truly random, since f is a TRF. Therefore, A will succeed with probability $\frac{1}{2}$. □

Lemma 5. $\Pr[\mathsf{Rep}] \leq \frac{t}{2^n}$, *where* $t = t_1 + t_2$.

Proof.

$$\begin{aligned}
&\Pr[\mathsf{Rep}] \\
&= \Pr\left[\bigcup_i (r_i = r)\right] \\
&\leq \sum_i \Pr[r = r_i] && \text{(by union bound)} \\
&= \sum_i 2^{-n} = \frac{t_1 + t_2}{2^n}
\end{aligned}$$

□

Lemma 6. $\Pr\limits_{f \in_R \mathsf{Func}_n}[D^f(1^n) = 1] \leq \frac{1}{2} + \frac{t}{2^n}$

Proof.

$$\begin{aligned}
&\Pr_{f \in_R \mathsf{Func}_n}[D^f(1^n) = 1] \\
&= \Pr_{f \in_R \mathsf{Func}_n}[b' = b] \\
&= \Pr\left[\mathrm{PrivK}^{\mathrm{cpa}}_{A,\widetilde{\Pi}}(n) = 1\right] \\
&= \Pr\left[\mathrm{PrivK}^{\mathrm{cpa}}_{A,\widetilde{\Pi}}(n) = 1 \;\middle|\; \mathsf{Rep}\right] \Pr[\mathsf{Rep}] + \Pr\left[\mathrm{PrivK}^{\mathrm{cpa}}_{A,\widetilde{\Pi}}(n) = 1 \;\middle|\; \overline{\mathsf{Rep}}\right] \Pr[\overline{\mathsf{Rep}}] \\
&\leq \Pr[\mathsf{Rep}] + \Pr\left[\mathrm{PrivK}^{\mathrm{cpa}}_{A,\widetilde{\Pi}}(n) = 1 \;\middle|\; \overline{\mathsf{Rep}}\right] \\
&\leq \frac{t}{2^n} + \frac{1}{2} && \text{(by lemma 5 and 4)}
\end{aligned}$$

□

$$\Pr_{f \in_R F}[D^f(1^n) = 1] - \Pr_{f \in_R \mathsf{Func}_n}[D^f(1^n) = 1]$$
$$\geq \left(\frac{1}{2} + \frac{1}{p(n)}\right) - \left(\frac{1}{2} + \frac{t}{2^n}\right)$$
$$= \frac{1}{p(n)} - \frac{t}{2^n} \notin \text{negl}(n) \qquad \text{(because } t \text{ is at most polynomial in } n\text{)}$$

Therefore, D is a successful distinguisher for F. □

2 Encrypting arbitrary length-messages

In the encryption scheme above, messages size is restricted to be equal to the input size of the PRF, $l_{\text{in}}(n)$. To encrypt variable length messages, we can simply break a long message into $l_{\text{in}}(n)$-length sub-messages and encrypt separately. Since Π is mult-CPA-secure, this way of breaking up messages is secure.

However, in this scheme, ciphertext size is twice the message size, which would be very inefficient when sending very large messages. There are other ways of encrypting arbitrary-length messages. These are called **block-cipher modes of operation**.

Reading assignment: Read about these block-cipher modes of operation:

- Electronic Code Book (ECB)
- Cipher Block Chaining (CBC)
- Output Feedback mode (OFB)
- Counter mode (CTR)

The following are some criteria for evaluating block-cipher modes of operation. Here n is the block-length and l is the number of blocks.

criterion	Π	Best-known
Randomness usage	ln	n
Ciphertext size	$2ln$	$(l+1)n$
Parallelizable encryption	Yes	Yes

3 PRG to PRF

3.1 Construction

Let $G : \mathbb{B}^n \mapsto \mathbb{B}^{2n}$ be a PRG. We will use it to construct a PRF $F : \mathbb{B}^n \times \mathbb{B}^n \mapsto \mathbb{B}^n$.

Let $G_0(x)$ be the first n bits of $G(x)$. Let $G_1(x)$ be the last n bits of $G(x)$.

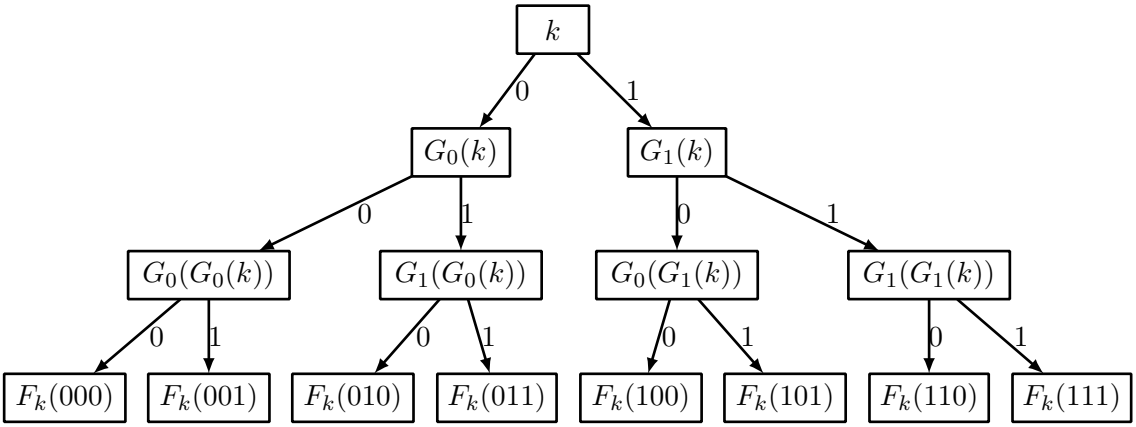

Figure 2: Definition of F_k: A complete binary tree is constructed using G and k. The leaf nodes represent the truth table of F_k.

Given k and G, we will construct a complete binary tree of depth n (see figure 2 for an example). The root node r will contain the value k. For every node v, let its left child be v_0 and its right child be v_1. Label the edge from v to v_i by i. If the value at v is x, then the value at v_i should be $G_i(x)$. Finally, $F_k(y)$ is defined as the value at the leaf node reached by following the path from the root given by the bits of y.

Since this tree has exponential size, this construction is infeasible. In fact, since the truth table of F_k is exponential in size, any construction which computes the whole truth table of F_k in one shot is infeasible.

Instead, we'll create the tree nodes 'lazily'. Initially the tree will only contain the root node. When we have to compute $F_k(y)$, we'll follow the path y from the root, creating only the missing nodes which are supposed to lie along the path (see figure 3). This way, to compute F_k for an input, we'll have to create at most n nodes.

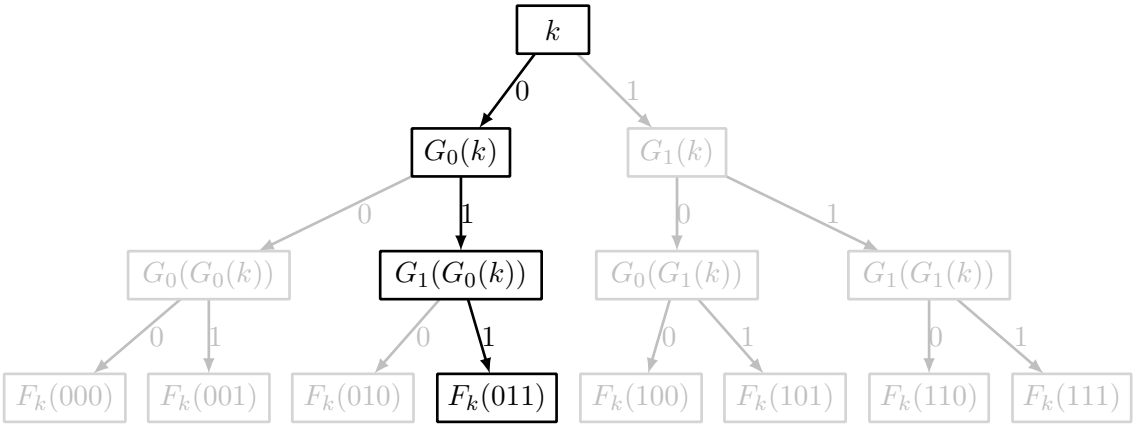

Figure 3: Computing $F_k(011)$: Only 3 nodes are created.

3.2 Stronger characterization of PRG

To prove that F is a PRF, we'll need to prove a property of PRGs. This property is a stronger generalization of the definition of PRG.

Theorem 7. *If G is a PRG then a PPT distinguisher cannot differentiate between a t-length vector of pseudorandom strings and a t-length vector of random strings. Formally,*

$$\left| \Pr_{s_1,\ldots,s_t \in_R \mathbb{B}^n}[A(G(s_1),\ldots,G(s_t)) = 1] - \Pr_{r_1,\ldots,r_t \in_R \mathbb{B}^{2n}}[A(r_1,\ldots,r_t) = 1] \right| \in \text{negl}(n)$$

Note that when $t = 1$, the theorem is trivially true by the definition of PRG.

Proof. The proof employs the Hybrid technique.

Define the probability distribution H_i as

$$H_i = \left\{ (G(s_1),\ldots,G(s_i), r_{i+1},\ldots,r_t) : (\forall j, s_j \in_R \mathbb{B}^n) \wedge (\forall j, r_j \in_R \mathbb{B}^{2n}) \right\}$$

We will prove that if A can distinguish between H_{i-1} and H_i, then there is a distinguisher for G.

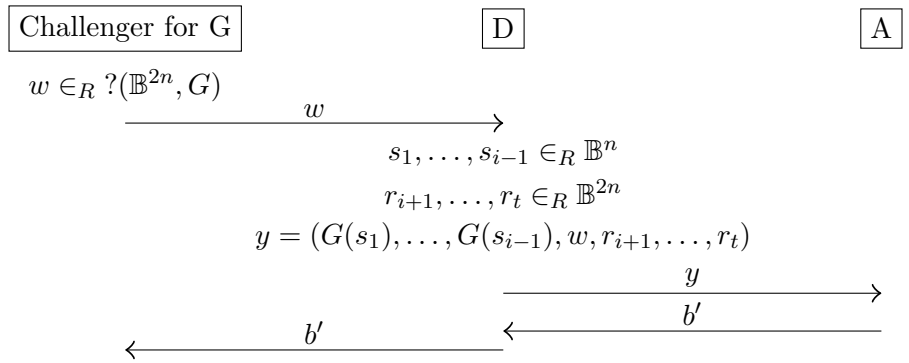

Figure 4: Constructing a distinguisher D for G which uses distinguisher A

Claim 8. *When $w \in_R \mathbb{B}^{2n}$, $y \in_R H_{i-1}$. When $w \in_R G$, $y \in_R H_i$.*

Lemma 9. *For any PPT distinguisher A, A cannot successfully distinguish H_{i-1} and H_i. Formally,*

$$\left| \Pr_{y \in H_i}[A(y) = 1] - \Pr_{y \in H_{i-1}}[A(y) = 1] \right| \in \text{negl}(n)$$

Proof. Assume that A is a successful distinguisher between H_{i-1} and H_i.

$$
\begin{aligned}
&\left| \Pr_{w \in G}[D(w) = 1] - \Pr_{w \in \mathbb{B}^{2n}}[D(w) = 1] \right| \\
&= \left| \Pr_{w \in G}[b' = 1] - \Pr_{w \in \mathbb{B}^{2n}}[b' = 1] \right| \\
&= \left| \Pr_{\substack{s_1,\ldots,s_{i-1},w \in G \\ r_{i+1},\ldots,r_t \in \mathbb{B}^{2n}}}[A(G(s_1),\ldots,G(s_{i-1}),w,r_{i+1},\ldots,r_t) = 1] \right. \\
&\quad \left. - \Pr_{\substack{s_1,\ldots,s_{i-1} \in G \\ w,r_{i+1},\ldots,r_t \in \mathbb{B}^{2n}}}[A(G(s_1),\ldots,G(s_{i-1}),w,r_{i+1},\ldots,r_t) = 1] \right| \\
&= \left| \Pr_{y \in H_i}[A(y) = 1] - \Pr_{y \in H_{i-1}}[A(y) = 1] \right| && \text{(by claim 8)} \\
&\notin \text{negl}(n) && (\because A \text{ can successfully distinguish } H_{i-1} \text{ and } H_i)
\end{aligned}
$$

This implies that G is not a PRG; a contradiction. Therefore, A cannot be a successful PPT distinguisher between H_{i-1} and H_i. □

$$
\begin{aligned}
&\left| \Pr_{s_1,\ldots,s_t \in_R \mathbb{B}^n}[A(G(s_1),\ldots,G(s_t)) = 1] - \Pr_{r_1,\ldots,r_t \in_R \mathbb{B}^{2n}}[A(r_1,\ldots,r_t) = 1] \right| \\
&= \left| \Pr_{y \in H_t}[A(y) = 1] - \Pr_{y \in H_0}[A(y) = 1] \right| \\
&= \left| \sum_{i=1}^{t} \left(\Pr_{y \in H_i}[A(y) = 1] - \Pr_{y \in H_{i-1}}[A(y) = 1] \right) \right| \\
&\leq \sum_{i=1}^{t} \left| \Pr_{y \in H_i}[A(y) = 1] - \Pr_{y \in H_{i-1}}[A(y) = 1] \right| \\
&\in \sum_{i=1}^{t} \text{negl}(n) && \text{(by lemma 9)} \\
&= \text{negl}(n) && \text{(sum of negligible functions is negligible)}
\end{aligned}
$$

□

3.3 Proof that F is a PRF

Theorem 10. *If G is a PRG, then the function F constructed as per section 3.1 is a PRF.*

The proof uses a hybrid argument.

Consider the distribution of all truth tables of F. Let's denote it by H_0. To sample from this distribution, generate $k \in_R \mathbb{B}^n$, set the root node's value to k and compute the value of

all other nodes in the tree. The values of all the leaf nodes together form a string of length $n2^n$ and this is the truth table for F_k.

Consider the distribution of all truth tables of truly random functions. Let's denote it by H_n. To sample a truly random function, directly set the leaf nodes to a uniformly random string of length $n2^n$. The values of all the leaf nodes together form a string of length $n2^n$ and this is the truth table for a random function.

We'll now consider a hybrid distribution H_i. To sample a function from this distribution, we'll set the i^{th} level of the tree to a uniformly random string of length $n2^i$ and compute the value of nodes in all levels from $i+1$ to n (the value of nodes in levels 0 to $i-1$ is `null`). The values of all the leaf nodes together form a string of length $n2^n$ and this is the truth table for a function chosen randomly from H_i. It is easy to see that this definition of H_i is consistent with the definitions of H_0 and H_n.

Once again, we need not compute the values of all nodes. We can do so lazily when we need to evaluate the function at an input x. Moreover, we also don't need to set the values of all nodes at the i^{th} level. When we need to evaluate the function at an input x, we'll first traverse the tree as per the path defined by the first i bits of x. During the traversal, we'll create nodes if they don't exist and the values of those nodes will be set to `null`. Suppose we reach a node v. If node v doesn't exist, we'll create it and set its value to a uniform random string. Then we'll traverse the tree from v to a leaf node by following the path defined by the rest of the bits of x. Along the path we'll create nodes which don't exist and compute their value using the value of their parent. The value at the leaf node will be the output of this function. This way of lazy evaluation enables us to compute the output of a random function from H_i in polynomial time.

Theorem 11. *For every PPT distinguisher A, A cannot successfully distinguish H_i from H_{i-1}. Formally,*

$$\left| \Pr_{f \in H_{i-1}}[A^f(1^n) = 1] - \Pr_{f \in H_i}[A^f(1^n) = 1] \right| \in \text{negl}(n)$$

Proof. Suppose a PPT distinguisher A can successfully distinguish H_i from H_{i-1}. We'll use A to construct a PPT distinguisher for G, contradicting the fact that G is a PRG.

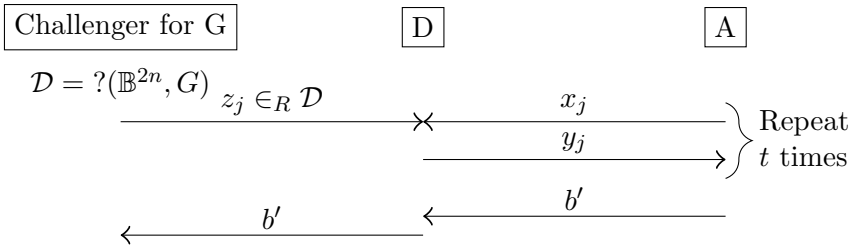

Figure 5: Constructing a distinguisher D for G which uses adversary A

D maintains a tree which initially has only the root node. In round j, D computes y_j as follows:

- Traverse tree from root as per first $i-1$ bits of x_j to reach node v (during traversal, create nodes which don't exist). Let the left and right children of v be v_0 and v_1 respectively. Note that v_0 and v_1 belong to the i^{th} layer of the tree.
- If v_0 and v_1 don't exist yet, create them. Set the value of v_0 to the left half of z_j and set the value of v_1 to the right half of z_j. If v_0 and v_1 already exist, don't modify their values.
- Resume traversal from v to a leaf as per the last $n-i+1$ bits of x_j. Create nodes which don't exist and compute their values using their parent.
- y_j is the value of the leaf node.

Lemma 12. *If $\mathcal{D} = \mathbb{B}^{2n}$, then A interacts with a function from H_i.*

Proof. If $\mathcal{D} = \mathbb{B}^{2n}$, then the i^{th} layer of the tree gets filled with uniform random values, which is the defining feature of functions in H_i. □

Lemma 13. *If $\mathcal{D} = G$, then A interacts with a function from H_{i-1}.*

Proof. Let $\mathcal{D} = G$. This means that $z_j = G(s_j)$ for $s_j \in_R \mathbb{B}^n$. Therefore, v_0 gets the value $G_0(s_j)$ and v_1 gets the value $G_1(s_j)$. This is equivalent to putting s_j at v (which is at level $i-1$) and computing values at v_0 and v_1 as per the tree construction. Since we're effectively putting uniform random strings at layer $i-1$, the function observed by A is from H_{i-1}. □

$$
\begin{aligned}
&\left| \Pr_{z_1,\ldots,z_t \in_R G}[D(z_1,\ldots,z_n)=1] - \Pr_{z_1,\ldots,z_n \in_R \mathbb{B}^{2n}}[D(z_1,\ldots,z_n)=1] \right| \\
&= \left| \Pr_{f \in_R H_{i-1}}[A^f(1^n)=1] - \Pr_{f \in_R H_i}[A^f(1^n)=1] \right| && \text{(by lemmas 13 and 12)} \\
&\notin \text{negl}(n) && (\because A \text{ can successfully distinguish } H_{i-1} \text{ and } H_i)
\end{aligned}
$$

By the contrapositive of the stronger characterization of PRGs (theorem 7), we get that G is not a PRG. This is a contradiction. Hence no PPT adversary can successfully distinguish H_{i-1} and H_i. □

$$
\begin{aligned}
&\left| \Pr_{f \in_R F}[A^f(1^n) = 1] - \Pr_{f \in_R \mathsf{Func}_n}[A^f(1^n) = 1] \right| \\
&= \left| \Pr_{f \in_R H_0}[A^f(1^n) = 1] - \Pr_{f \in_R H_n}[A^f(1^n) = 1] \right| \\
&= \left| \sum_{i=1}^{n} \left(\Pr_{f \in_R H_{i-1}}[A^f(1^n) = 1] - \Pr_{f \in_R H_i}[A^f(1^n) = 1] \right) \right| \\
&\leq \sum_{i=1}^{n} \left| \Pr_{f \in_R H_{i-1}}[A^f(1^n) = 1] - \Pr_{f \in_R H_i}[A^f(1^n) = 1] \right| \\
&\in \sum_{i=1}^{t} \mathrm{negl}(n) && \text{(by theorem 11)} \\
&= \mathrm{negl}(n) && \text{(sum of negligible functions is negligible)}
\end{aligned}
$$

Therefore, F is a PRF.