Security of the Counter Mode

In Figure 1 we describe the randomized counter mode. The following notations have been used:

For $x \in \{0,1\}^*$, $\mathsf{parse}_n(x)$ returns $x_1||x_2||\dots||x_m$, where $m = \lceil |x|/n \rceil$, and for $1 \le i \le m-1$, $|x_i| = n$ and $0 < |x_m| \le n$.
For a non-negative integer $i < 2^n - 1$, $\mathsf{bin}_n(i)$ denote the n bit binary representation of i.
For $x \in \{0,1\}^*$ and $|x| \ge n$, $\mathsf{Take}_n(x)$ returns the first n bits of x.

$\mathsf{CTR}_K(M)$	$\mathsf{CTR}_K^{-1}(C)$
1. $M_1||M_2||\dots||M_m \leftarrow \mathsf{parse}_n(M)$;	1. $C_0||C_1||C_2||\dots||C_m \leftarrow \mathsf{parse}_n(C)$;
2. $IV \xleftarrow{\$} \{0,1\}^n$;	2. $IV \leftarrow C_0$;
3. **for** $i = 1$ to $m-1$;	3. **for** $i = 1$ to $m-1$;
4. $\quad z_i \leftarrow E_K(IV \oplus \mathsf{bin}_n(i))$;	4. $\quad z_i \leftarrow E_K(IV \oplus \mathsf{bin}_n(i))$;
5. $\quad C_i \leftarrow z_i \oplus M_i$;	5. $\quad M_i \leftarrow z_i \oplus C_i$;
6. **end for** ;	6. **end for** ;
7. $C_m \leftarrow \mathsf{Take}_{\lvert M_m \rvert}(E_K(IV \oplus m)) \oplus M_m$;	7. $M_m \leftarrow \mathsf{Take}_{\lvert C_m \rvert}(E_K(IV \oplus \mathsf{bin}_n(m))) \oplus C_m$;
8. **return** $IV; C_1||C_2||\dots||C_m$;	8. **return** $IV; M_1||M_2||\dots||M_m$;

Fig. 1. The randomized Counter mode.

Theorem 1. *Let $E : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a function family and let* CTR *be the symmetric encryption algorithm as described in Figure 1. Let $\mathcal{A}$ be an arbitrary adversary attacking* CTR *in the* IND$ *sense, who asks at most q queries, these totaling at most σ n-bit blocks. Then there exists an adversary $\mathcal{B}$ such that*

$$\mathbf{Adv}^{\mathrm{ind\$}}_{\mathsf{CTR}}(\mathcal{A}) \le \mathbf{Adv}^{\mathrm{prf}}_{E}(\mathcal{B}) + \frac{\sigma^2}{2^{n+1}}. \tag{1}$$

And if $\mathcal{A}$ runs in time at most t then $\mathcal{B}$ runs in time at most $t' = t + O(q)$ and asks at most $q' = \sigma$ oracle queries.

Before we give the proof, it would be worthwhile to analyze a bit the statement of the Theorem. The Theorem states that the IND$ advantage of an arbitrary adversary in breaking CTR is upper bounded by the sum of $\sigma^2/2^{n+1}$ and the PRF advantage of some adversary $\mathcal{B}$ in breaking E_K. If we assume that E is a PRF-family then for any "efficient" adversary $\mathcal{B}$, $\mathbf{Adv}^{\mathrm{prf}}_{E}(\mathcal{B})$, is bound to be small. Thus, if σ^2 is negligibly small compared to 2^n, then the Theorem guarantees that any adversary will have a small IND$ advantage in breaking CTR. Thus, the Theorem is a relative statement regarding the security of CTR. It guarantees CTR to be IND$ secure if the following holds:

– E is a pseudorandom function family.
– σ^2 is negligibly small compared to 2^n. This condition basically says that CTR is not secure, if an unlimited number of queries is asked by the adversary. Also, this bound may be useful in deciding how many plaintexts are to be encrypted under the same key.

Remark 1. We discussed in class that the security of a block cipher is that of a pseudorandom-permutation, but here we are assuming the block cipher to be a PRF. First note, that in CTR no call of the inverse block cipher is required (even for decryption), thus for CTR to function it is not required that E is a permutatoion. But we would generally instantiate CTR with a block cipher, which is a permutation, we can choose E to be a permutation family and use the PRP-PRF switching lemma to replace $\mathbf{Adv}^{\mathrm{prf}}_E(\mathcal{B})$ in eq. (1) by $\mathbf{Adv}^{\mathrm{prp}}_E(\mathcal{B}) + q(q-1)/2^{n+1}$.

Now we proceed to prove Theorem 1.

Proof. Let $\mathcal{A}$ be an adversary attacking CTR in the IND\$ sense. Now we will construct a PRF adversary $\mathcal{B}$ which will respond to the queries of $\mathcal{A}$. We describe $\mathcal{B}$ in Figure 2. $\mathcal{B}$ being a PRF adversary has access to an oracle $\mathcal{O}$ which can be either $E_K()$, for $K \xleftarrow{\$} \mathcal{K}$, or a function $\rho \xleftarrow{\$} \mathsf{Func}(n, n)$.

Adversary $\mathcal{B}^{\mathcal{O}}$
1. Whenever $\mathcal{B}$ gets a query M from $\mathcal{A}$
2. **do** the following until $\mathcal{A}$ stops querying
3. $\quad M_1||M_2||\dots||M_m \leftarrow \mathsf{parse}_n(M)$;
4. $\quad IV \xleftarrow{\$} \{0,1\}^n$;
5. $\quad$ **for** $i = 1$ to $m-1$;
6. $\quad\quad z_i \leftarrow \mathcal{O}(IV \oplus \mathsf{bin}_n(i))$;
7. $\quad\quad C_i \leftarrow z_i \oplus M_i$;
8. $\quad$ **end for** ;
9. $\quad C_m \leftarrow \mathsf{Take}_{|M_m|}(\mathcal{O}(IV \oplus m)) \oplus M_m$;
10. $\quad$ **return** $(IV; C_1||C_2||\dots||C_m)$ to $\mathcal{A}$;
11. Finally $\mathcal{A}$ returns a bit b;
12. $\mathcal{B}$ returns b;

Fig. 2. Adversary $\mathcal{B}$ for the proof of Theorem 1.

In the description of $\mathcal{B}$, lines 3 to 10 denotes the encryption function of CTR shown in Figure 1, where $E_K()$ is replaced by $\mathcal{O}$. Note, $\mathcal{O}$ is the oracle of $\mathcal{B}$. The goal of $\mathcal{B}$ is to guess if $\mathcal{O}$ is $E_K()$ or a random function ρ. $\mathcal{B}$ tries to reach this goal, with the help of $\mathcal{A}$.

Note, that if the oracle $\mathcal{O}$ is E_K, then $\mathcal{B}$ provides the outputs of CTR to $\mathcal{A}$, and it is clear that

$$\Pr\left[K \xleftarrow{\$} \mathcal{K} : \mathcal{B}^{E_K(.)} \Rightarrow 1\right] = \Pr\left[K \xleftarrow{\$} \mathcal{K} : \mathcal{A}^{\mathsf{CTR}(.)} \Rightarrow 1\right]. \tag{2}$$

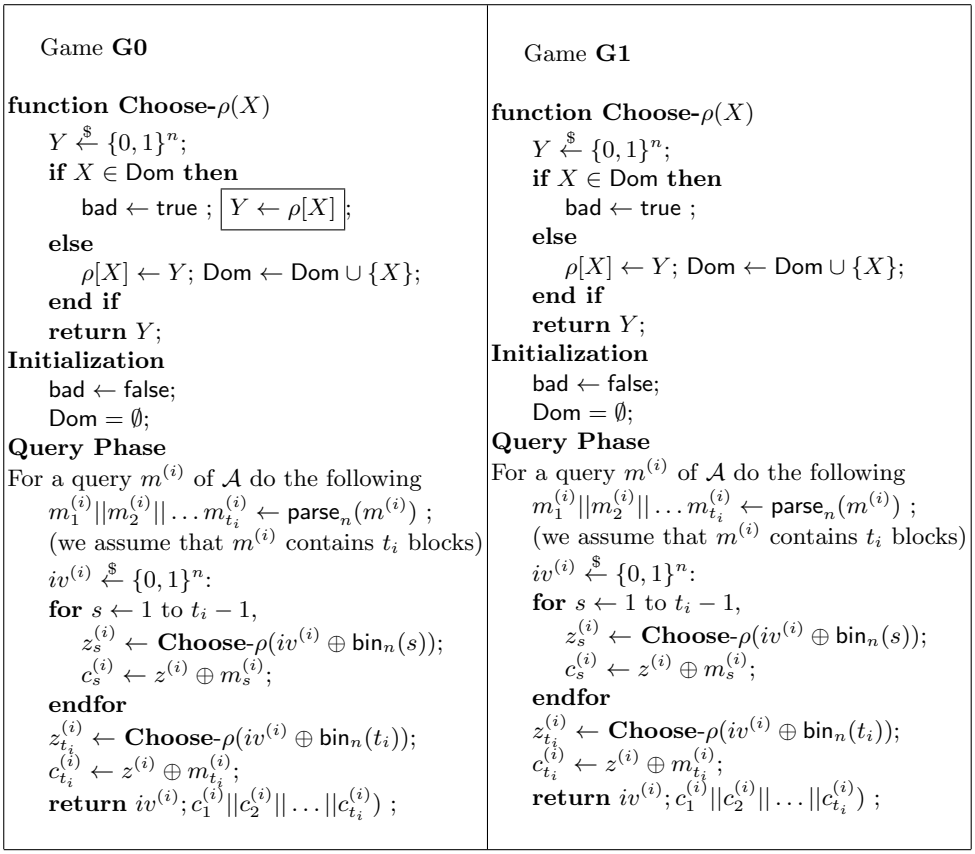

Fig. 3. Games **G0, G1, G2** used for the proof of Theorem 1.

Now, we would like to analyze the situation where $\mathcal{O}$ is a random function. To do this, we use the technique of sequence of games. Consider the game **G0** shown in Figure 3. The game **G0** includes a function **Choose**-$\rho()$, which acts as a random function. It returns uniform random strings in $\{0,1\}^n$ when it is invoked, but it returns the same string if invoked twice on the same input. It does this by maintaining a table ρ of outputs that it has already returned. Additionally in the set Dom, it maintains the points on which it has been queried. The function sets the bad flag to true if it is queried twice on the same input. The game **G0** assumes that the ith query of the adversary is a message containing t_i n-bit blocks, and the challenge query contains messages of t blocks.

The game **G0** does the same as $\mathcal{B}$ does where the oracle of $\mathcal{B}$ is replaced by the function **Choose**-$\rho()$. As **Choose**-ρ acts like a random function, hence it is immediate that

$$\Pr[\rho \xleftarrow{\$} \mathsf{Func}(n) : \mathcal{B}^{\rho(\cdot)} \Rightarrow 1] = \Pr[\mathbf{G0} \Rightarrow 1] \tag{3}$$

Now, we do a small change in game **G0**, i.e., we remove the boxed entry in the function **Choose**-ρ, we call this changed game as **G1**. Notice that games **G1** and **G0** are identical until the flag bad is set to true, hence by the difference lemma we have

$$\left|\Pr[\mathcal{A}^{\mathbf{G0}} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathbf{G1}} \Rightarrow 1]\right| \leq \Pr[\mathcal{A}^{\mathbf{G1}} \text{ sets } \mathsf{bad}] \tag{4}$$

Also in game **G1**, the function **Choose**-ρ, returns random strings for any input it gets, thus $\mathcal{A}$ when interacts with **G1** gets random strings of size of the queried plaintexts as response to its queries. Hence,

$$\Pr[\mathcal{A}^{\mathbf{G1}} \Rightarrow 1] = \Pr[\mathcal{A}^{\$} \Rightarrow 1]. \tag{5}$$

Now, we have

$$\begin{aligned}
\mathbf{Adv}^{\mathrm{ind\$}}_{\mathsf{CTR}}(\mathcal{A}) &= \left|\Pr\left[K \xleftarrow{\$} \mathcal{K} : A^{\mathsf{CTR}} \Rightarrow 1\right] - \Pr\left[\mathcal{A}^{\$} \Rightarrow 1\right]\right| && (6)\\
&= \left|\Pr\left[K \xleftarrow{\$} \mathcal{K} : \mathcal{B}^{E_K(.)} \Rightarrow 1\right] - \Pr[\mathcal{A}^{\mathbf{G0}} \Rightarrow 1] + \Pr[\mathcal{A}^{\mathbf{G0}} \Rightarrow 1] - \Pr[\mathcal{A}^{\$} \Rightarrow 1]\right| && (7)\\
&= \Big|\Big(\Pr\left[K \xleftarrow{\$} \mathcal{K} : \mathcal{B}^{E_K(.)} \Rightarrow 1\right] - \Pr\left[\rho \xleftarrow{\$} \mathsf{Func}(n,n) : \mathcal{B}^{\rho(.)} \Rightarrow 1\right]\Big) && (8)\\
&\quad + \Big(\Pr[\mathcal{A}^{\mathbf{G0}} \Rightarrow 1] - \Pr[\mathcal{A}^{\$} \Rightarrow 1]\Big)\Big| && (9)\\
&\leq \mathbf{Adv}^{\mathrm{prf}}_{E}(\mathcal{B}) + \left|\Pr[\mathcal{A}^{\mathbf{G0}} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathbf{G1}} \Rightarrow 1]\right| && (10)\\
&\leq \mathbf{Adv}^{\mathrm{prf}}_{E}(\mathcal{B}) + \Pr[\mathcal{A}^{\mathbf{G1}} \text{ sets } \mathsf{bad}] && (11)
\end{aligned}$$

To complete the proof we need to bound $\Pr[\mathcal{A}^{\mathbf{G1}} \text{ sets } \mathsf{bad}]$. If we consider Dom in **G1** to be a multiset, then the event $\mathcal{A}^{\mathbf{G1}}$ sets bad is same as the event that there are two elements in Dom with the same value. We call this event as a collision in Dom. Let COLLD denote the event that there is a collision in the multiset Dom in game **G2**, then from the description of game **G2**, we have

$$\Pr[\mathbf{G2} \text{ sets } \mathsf{bad}] = \Pr[\mathsf{COLLD}]. \tag{12}$$

Now we concentrate on finding an upper bound for $\Pr[\mathsf{COLLD}]$. The elements present in Dom are given by

$$\mathsf{Dom} = \bigcup_{i=1}^{q} \{iv^{(i)} \oplus \mathsf{bin}_n(1), iv^{(i)} \oplus \mathsf{bin}_n(2), \ldots, iv^{(i)} \oplus \mathsf{bin}_n(t_i)\}.$$

Claim. Let $x, y \in \mathsf{Dom}$, then $\Pr[x = y] \leq 1/2^n$.

Note that each element in Dom is of the form $iv^i \oplus \mathsf{bin}_n(j)$. Let $x = iv^{(i_1)} \oplus \mathsf{bin}_n(j_1)$ and $y = iv^{(i_2)} \oplus \mathsf{bin}_n(j_2)$. We consider the following two cases:

Case 1: $i_1 = i_2$. If $i_1 = i_2$ then it is always the case that $j_1 \neq j_2$, which makes $\Pr[x = y] = 0$.
Case 2:$i_1 \neq i_2$. In this case $iv^{(i_1)}$ and $iv^{(i_2)}$ are two uniform and independent random elements in $\{0,1\}^n$, thus making $\Pr[x = y] = 1/2^n$.

From the above two cases the claim follows.

Also, it is easy to see that Dom contains $\sum_{i=1}^{q} t_i$ elements, which is same as the total number of $n - bit$ blocks of queries σ_n made by the adversary. Hence, by the union bound we have

$$\Pr[\mathsf{COLLD}] = \binom{\sigma}{2}\frac{1}{2^n} = \frac{\sigma(\sigma-1)}{2^{n+1}} \leq \frac{\sigma^2}{2^{n+1}}. \tag{13}$$

Thus using equations (11), (12) and (13) we have

$$\mathbf{Adv}^{\mathrm{ind\$}}_{\mathsf{CTR}}(\mathcal{A}) \leq \mathbf{Adv}^{\mathrm{prf}}_{E}(\mathcal{B}) + \frac{\sigma^2}{2^{n+1}},$$

as desired. □