Solution of Exercisesheet 8

Exercise 8-1

Lets assume towards contradiction that the MAC construction is not secure $\Rightarrow$ Probablity of forging this construction Π_{MAC} is a non negligible function:

$$Pr[MacForge_{A,\Pi_{MAC}} = 1] \leq \epsilon(\lambda)$$

where $\epsilon(\lambda)$ is a non negligible function.

That implies there exists an efficient adversary, A, able to generate a new message and tag pair (m^*, t^*) such that $m^* \notin Q$, and $Ver_k(m^*, t^*) == 1$ with a probablity $\epsilon(\lambda)$.

We now consider $\mathcal{B}$ attacking the CCA-securitv. $\mathcal{B}$ runs $\mathcal{A}$ as subroutine. $\mathcal{B}$ only forward all encryption queries that $\mathcal{A}$ asks for to his own encryption oracle. Finally $\mathcal{A}$ outputs a message-tag-pair m^*, t^*. $\mathcal{B}$ then chooses two messages, lets say $\widetilde{m_0}$ as m^* and $\widetilde{m_1}$ as any other random message. If $\mathcal{B}$ gets the tag t^* as $\widetilde{c_b}$, it corresponds to $m_0 = m^*$ otherwise it correponds to random message m_1. This holds, because $Dec_k(t) = m \Leftrightarrow Enc_k(m) = t$.
$\mathcal{B}$ is efficient because it only forwards messages, which are of polynomial length (because $\mathcal{A}$ is efficient), chooses a random number and invokes $\mathcal{A}$. Because $\mathcal{A}$ can break the MAC with non-negligible probability $\epsilon(\lambda)$ and $\mathcal{B}$ uses this in every case, $\mathcal{B}$ can break the CCA security also with non-negligible probability $\epsilon(\lambda)$. Because this a contradiction to the CCA security of Π, our assumption that such an adversary $\mathcal{A}$ against the collision resistance of Π_{MAC} exists, is false. So Π_{MAC} is a collision resistant MAC.

Exercise 8-2

(a) Assume H is not collision resistant.
This means finding $x_0 \neq x_1$ so that $H^{s_0||s_1}(x_0) = H^{s_0||s_1}(x_1)$ is possible with non-negl. probability.
$H^{s_0||s_1}(x_0) = H^{s_0||s_1}(x_1) \Leftrightarrow H_0^{s_0}(H_1^{s_1}(x_0)) = H_0^{s_0}(H_1^{s_1}(x_1))$
Define $H_1^{s_1}(x_0) = y_0$ and $H_1^{s_1}(x_1) = y_1$.
For $H^{s_0||s_1}(x_0) = H^{s_0||s_1}(x_1)$ to have a collision either $y_0 = y_1 = y$
which makes $H_0^{s_0}(y_0) = H_0^{s_0}(y_1) = H_0^{s_0}(y)$ trivially true because $H_0^{s_0}$ gets the same input
or $H_0^{s_0}(y_0) = H_0^{s_0}(y_1)$ with $y_0 \neq y_1$

If H_0 is collision resistant but H_1 not, then the probability of finding $x_0 \neq x_1$ so that $H_1^{s_1}(x_0) = H_1^{s_1}(x_1) = y$ is non-negl.
Since H_0 gets the same input y in this case the probability for a collision for H is the same as for H_1 and is non-negl.
$\Rightarrow H$ is not collision resistant.

(b) Assume H is not collision resistant.
This means finding $x_0 \neq x_1$ so that $H^{s_0||s_1}(x_0) = H^{s_0||s_1}(x_1)$ is non-negl.
$H^{s_0||s_1}(x_0) = H^{s_0||s_1}(x_1) \Leftrightarrow H_0^{s_0}(x_0)||H_1^{s_1}(x_0) = H_0^{s_0}(x_1)||H_1^{s_1}(x_1)$

Solution of Exercisesheet 8

We know that at least one H_i $i \in \{0,1\}$ is collision resistant per definition.

Assume H_0 is collision resistant but H_1 is not, making finding $x_0 \neq x_1$ so that $H_0^{s_0}(x_0) = H_0^{s_0}(x_1)$ is negl.
If $H_1^{s_1}(m) = c$ with c being a constant value for all m then $H^{s_0||s_1}(m) = H_0^{s_0}(m)||c$
This produces a string with the same collision resistance propability as H_0 since only a constant string c is added at the end.
If H_1 is some other function, the collision probability becomes even smaller since there are even more different strings produceable this way.

Assuming H_1 is collision resistant but H_0 is not, has the same result since it doesn't matter if the collision resistant part is added at first or second.

$\Rightarrow$ contradiction: H can't be non-negl. since it has to have at least the collision resistant propablility as the collision resistant H_i $i \in \{0,1\}$ which is negl.
$\Rightarrow$ H is collision resistant.

(c) $H'(m) = H^{c(m)}(r(m))$ with m of the format $0^n1||x$ and $c(m) = n$ and $r(m) = x$.
Since H^n is the n-times application of H we can construct the following attack on this Hash function to break the collision resistance:
$H^n(x) = y_0$ as one random example and
$H^1(y_0) = y_1$
We use $x_0 = 0^{n+1}1||x$ and $x_1 = 01||y_0$ since
$H'(x_0) = H^{c(x_0)}(r(x_0)) = H^{n+1}(x) = H^1(H^n(x)) = H^1(y_0) = H^{c(x_1)}(r(x_1)) = H'(x_1)$.
$\Rightarrow$ H is not collision resistant.

Exercise 8-3

$F_k(x) := H(k \parallel x)$ is not a pseudorandom function.
Proof:
Let's assume the fixed-length collision resistant hash function (Gen, h) that is used in the Merkle-Damgard construction always concatenates a 1 at the beginning of every output. This property is perfectly legal for a fixed-length collision resistant hash function.
For each step in the Merkle-Domgard construction, the output will therefore have a 1 at the beginning as each $z_i := h^s(z_{i-1} \parallel x_i)$.
Therefore, $F_k(x) := H(k \parallel x)$ will also always have its output starting with a 1.
$F_k(x)$ can easilily be distinguished from a uniformly selected function f. In polynomial time, a distinguisher can easily determine that $F_k(x)$ is not truely random as 1 always being the first bit is an obvious hint.

Solution of Exercisesheet 8

Exercise 8-4

To show: $H(m) : \{0,1\}^{2k} \to \{0,1\}^{k+n}, H(m) := m_0||H'(m_1)$ is still a collision-resistant hash function when $m = m_0||m_1, |m_0| = |m_1| = k$ and $k > n$. $H'(m) : \{0,1\}^* \to \{0,1\}^n$ is a collision-resistant hash function.

Proof by contradiction. We assume there is an adversary $\mathcal{A}$, who can break the collision-resistance of $H(m)$ with non-negligible probability. We now build an adversary $\mathcal{B}$ against the collision-resistance of $H'(m)$ who invokes $\mathcal{A}$. $\mathcal{A}$ then outputs two messages m^1, m^2. $\mathcal{B}$ computes his output by truncating the first half of m^1 and m^2 ($m^i = m_0^i||m_1^i, i \in \{1,2\}$).

$\mathcal{B}$ is an efficient adversary because $\mathcal{A}$ is efficient, so the message length is poly and the call to $\mathcal{A}$ needs only poly time and sampling and prepend m_0 and truncating bit from m^1 and m^2 can also be done in polynomial time.

To analyse the success, we know, that with non-negligible probability $\mathcal{A}$ outputs two messages m^1, m^2 with $m^1 \neq m^2$ and $H(m^1) = H(m^2)$. $\mathcal{B}$ outputs only the second half of m^1 and m^2 which results in m_1^1, m_1^2. The probability that these are equal is $\left(\frac{1}{2}\right)^n$, because for each position the probability that the bits are equal is $\frac{1}{2}$. In all other cases $\mathcal{B}$ outputs two messages m_1^1, m_1^2 with $m_1^1 \neq m_1^2$ and $H'(m_1^1) = H'(m_1^2)$. This holds because $H(m^1) = H(m^2) \Rightarrow H(m_0^1||m_1^1) = H(m_0^2||m_1^2) \Rightarrow m_0^1||H'(m_1^1) = m_0^2||H'(m_1^2)$.

$$Pr[HashColl_{\mathcal{B}}(\lambda) = 1] = Pr[HashColl_{\mathcal{A}}(\lambda) = 1] - Pr[x_1' == x_2'] = \texttt{non-negl.} - \left(\frac{1}{2}\right)^k = \texttt{non-negl.}$$

Because this is a contradiction to the collision-resistance of $H'(m)$ such an adversary $\mathcal{A}$ cannot exist.

It follows that $H(m)$ is a collision-resistant hash function.