Exercise (HASH + MAC = MAC). Let $\mathcal{H}$ be a collision resistant hash function family form $\mathcal{M}$ to $\mathcal{X}$ and let $\text{Mac}: \mathcal{X} \times \mathcal{K} \rightarrow \mathcal{T}$ be a secure message authentication code. Show that the following function

$$\text{HashMac}(m, k, h) = \text{Mac}(h(m), k)$$

is secure message message authentication code with a signature $\text{HashMac} : \mathcal{M} \times \mathcal{K} \times \mathcal{H} \rightarrow \mathcal{T}$, i.e., the usage of collision resistant functions allows us to extend the domain of a message authentication code.

Solution. Recall that according to the security definition for message authentication we must show that the probability that a $t$-time adversary $A$ wins the following game

$$G_A^A$$

is bounded from above. By substituting the definition of $\text{HashMac}$ into the game, we obtain

$$G_0^A$$

is bounded from above. By substituting the definition of $\text{HashMac}$ into the game, we obtain

Note that $A$ wins the game, if $A$ creates $m$ such that $h(m) = h(m_i)$ while $m \neq m_i$. Then $t_i$ is a known and
valid message authentication tag for \( m \). To handle this issue explicitly, we can define the following games:

\[ G_1^A \]
\[
\begin{align*}
k &\leftarrow \mathcal{K} \\
h &\leftarrow \mathcal{H} \\
t_0 &\leftarrow A(h) \\
\text{For } i \in \{1, \ldots, q\} &\text{ do} \\
m_i &\leftarrow A(t_{i-1}) \\
x_i &\leftarrow h(m_i) \\
t_i &\leftarrow \text{Mac}(x_i, k) \\
(m, t) &\leftarrow A(t_q) \\
\text{if } [h(m) \notin \{h(m_1), \ldots, h(m_q)\}] &\text{ return } 0 \\
\text{if } m \in \{m_1, \ldots, m_q\} &\text{ return } 0 \\
\text{return } [t \triangleq \text{Mac}(h(m), k)]
\end{align*}
\]

\[ G_2^A \]
\[
\begin{align*}
k &\leftarrow \mathcal{K} \\
h &\leftarrow \mathcal{H} \\
t_0 &\leftarrow A(h) \\
\text{For } i \in \{1, \ldots, q\} &\text{ do} \\
m_i &\leftarrow A(t_{i-1}) \\
x_i &\leftarrow h(m_i) \\
t_i &\leftarrow \text{Mac}(x_i, k) \\
(m, t) &\leftarrow A(t_q) \\
\text{if } h(m) \in \{h(m_1), \ldots, h(m_q)\} &\text{ return } 0 \\
\text{return } [t \triangleq \text{Mac}(h(m), k)]
\end{align*}
\]

Clearly, we can split all runs of \( G^A \) into two classes depending whether the event \( h(m) \notin \{h(m_1), \ldots, h(m_q)\} \) holds or not. As the event \( h(m) \notin \{h(m_1), \ldots, h(m_q)\} \) also implies \( m \notin \{m_1, \ldots, m_q\} \), we do not have to check the condition \( (m, t) \in \{(m_1, t_1), \ldots, (m_q, t_q)\} \) any more in \( G_2 \). For the game \( G_1 \), we still have to check that \( m \notin \{m_1, \ldots, m_q\} \). Thus, by the construction of games we have established

\[
\Pr [G_0^A = 1] = \Pr [G_1^A = 1] + \Pr [G_2^A = 1].
\]

The game \( G_2 \) is very close to the security game for the message authentication codes. In fact, if we define an adversary \( B \) such that

\[ B(t_0) \]
\[
\begin{align*}
h &\leftarrow \mathcal{H} \\
m_1 &\leftarrow A(t_0) \\
\text{return } h(m_1)
\end{align*}
\]

\[ B(t_{i-1}) \]
\[
\begin{align*}
m_i &\leftarrow A(t_{i-1}) \\
\text{return } h(m_i)
\end{align*}
\]

\[ B(t_q) \]
\[
\begin{align*}
(m, t) &\leftarrow A(t_0) \\
\text{return } h(m_1), t
\end{align*}
\]

then direct substitution to the security game of message authentication code

\[ Q^B \]
\[
\begin{align*}
k &\leftarrow \mathcal{K} \\
\text{For } i \in \{1, \ldots, q\} &\text{ do} \\
x_i &\leftarrow B(t_{i-1}) \\
t_i &\leftarrow \text{Mac}(x_i, k) \\
(x, t) &\leftarrow A(t_q) \\
\text{if } (x, t) \in \{(x_1, t_1), \ldots, (x_q, t_q)\} &\text{ return } 0 \\
\text{return } [t \triangleq \text{Mac}(x, k)]
\end{align*}
\]

leads to the game that is equivalent to the game \( G_2^A \). The only difference after the mechanical substitution is in the last check. In the game \( G_2^A \), the check

\[
h(m) \in \{h(m_1), \ldots, h(m_q)\}
\]

is more stringent than the check \( (h(m), t) \in \{(h(m_1), t_1), \ldots, (h(m_q), t_q)\} \) used in \( Q^B \). Consequently,

\[
\Pr [G_2^A = 1] \leq \Pr [Q^B = 1] \leq \text{Adv}_{\text{Mac}}^B(h).
\]
Note that the overhead in the running time of $\mathcal{B}$ is linear in the number of queries $q$ and thus $(t + O(q), \varepsilon_1)$-secure message authentication code is sufficient for bounding the success probability in the game $\mathcal{G}_2$.

For the game $\mathcal{G}_1$, it is important to note that $A$ passes first two checks only if $A$ creates a hash collision: $h(m) = h(m_i)$ for $m \neq m_i$. Consequently, the following adversary

$$
\mathcal{C}(h)
$$

\begin{align*}
& k \leftarrow K \\
& t_0 \leftarrow A(h) \\
& \text{For } i \in \{1, \ldots, q\} \text{ do} \\
& \quad m_i \leftarrow A(t_{i-1}) \\
& \quad x_i \leftarrow h(m_i) \\
& \quad t_i \leftarrow \text{Mac}(x_i, k) \\
& (m_i, t_i) \leftarrow A(t_q) \\
& \text{if } [h(m) \notin \{h(m_1), \ldots, h(m_q)\}] \text{ return } 0 \\
& i \leftarrow \{i : h(m_i) = h(m)\} \\
& \text{return } (m, m_i)
\end{align*}

can be used for the collision resistance game $\mathcal{Q}^C$

$$
\mathcal{Q}^C
$$

\begin{align*}
& h \leftarrow \mathcal{H} \\
& (m_0, m_1) \leftarrow \mathcal{B}(h) \\
& \text{return } [h(m_0) = h(m_1)] \land [m_0 \neq m_1]
\end{align*}

Again, the success criterion in the game $\mathcal{Q}$ is more relaxed than in the game $\mathcal{G}_1$ and thus direct substitution allows us to prove:

$$
\Pr[\mathcal{G}_1^A = 1] \leq \Pr[\mathcal{Q}^C = 1] \leq \text{Adv}_{\mathcal{H}}^\varepsilon(\mathcal{C}) .
$$

Again, the overhead in the running time of $\mathcal{C}$ is $O(q)$. Thus, usage of $(t + O(q), \varepsilon_2)$-collision resistant hash function family $\mathcal{H}$ is sufficient for bounding the success probability in the game $\mathcal{G}_1$.

To summarise, we have proven that HashMac is $(t, q, \varepsilon_1 + \varepsilon_2)$-secure message authentication code provided that $\mathcal{H}$ is a $(t + O(q), \varepsilon_2)$-collision resistant hash function family and Mac is $(t + O(q), \varepsilon_1)$-secure message authentication code.

On the optimality of bounds. It is easy to see that $A$ can win $\mathcal{G}_1$ as soon as it produces a hash collision $h(m) = h(m_i)$, since $A$ can set $t = t_i$ and pass the last check, as well. Most message authentication codes are deterministic and thus the conditions

$$
h(m) \in \{h(m_1), \ldots, h(m_q)\} \quad \text{and} \quad (h(m), t) \in \{(h(m_1), t_1), \ldots, (h(m_q), t_q)\}
$$

are equivalent. The latter implies that also the second bound is optimal.