Domain Extension for MACs
CBC-MAC

Let $F$ be a pseudorandom function, and fix a length function $\ell$. The basic CBC-MAC construction is as follows:

- **Mac**: on input a key $k \in \{0,1\}^n$ and a message $m$ of length $\ell(n) \cdot n$, do the following:
  1. Parse $m$ as $m = m_1, \ldots, m_\ell$ where each $m_i$ is of length $n$.
  2. Set $t_0 := 0^n$. Then, for $i = 1$ to $\ell$:
     ```
     \text{Set } t_i := F_k(t_{i-1} \oplus m_i).
     ```
     Output $t_\ell$ as the tag.

- **Vrfy**: on input a key $k \in \{0,1\}^n$, a message $m$, and a tag $t$, do: If $m$ is not of length $\ell(n) \cdot n$ then output 0. Otherwise, output 1 if and only if $t = Mac_k(m)$. 
CBC-MAC

FIGURE 4.1: Basic CBC-MAC (for fixed-length messages).
Chosen Ciphertext Security
CCA Security

The CCA Indistinguishability Experiment $PrivK_{A,\Pi}^{cca}(n)$:

1. A key $k$ is generated by running $Gen(1^n)$.
2. The adversary $A$ is given input $1^n$ and oracle access to $Enc_k(\cdot)$ and $Dec_k(\cdot)$, and outputs a pair of messages $m_0, m_1$ of the same length.
3. A random bit $b \leftarrow \{0,1\}$ is chosen, and then a challenge ciphertext $c \leftarrow Enc_k(m_b)$ is computed and given to $A$.
4. The adversary $A$ continues to have oracle access to $Enc_k(\cdot)$ and $Dec_k(\cdot)$, but is not allowed to query the latter on the challenge ciphertext itself. Eventually, $A$ outputs a bit $b'$.
5. The output of the experiment is defined to be 1 if $b' = b$, and 0 otherwise.
CCA Security

A private-key encryption scheme $\Pi = (Gen, Enc, Dec)$ has indistinguishable encryptions under a chosen-ciphertext attack if for all ppt adversaries $A$ there exists a negligible function $negl$ such that

$$\Pr \left[ PrivK_{A,\Pi}^{cca}(n) = 1 \right] \leq \frac{1}{2} + negl(n),$$

where the probability is taken over the random coins used by $A$, as well as the random coins used in the experiment.
Authenticated Encryption

The unforgeable encryption experiment $\text{EncForge}_{A,\Pi}(n)$:

1. Run $\text{Gen}(1^n)$ to obtain key $k$.

2. The adversary $A$ is given input $1^n$ and access to an encryption oracle $\text{Enc}_k(\cdot)$. The adversary outputs a ciphertext $c$.

3. Let $m := \text{Dec}_k(c)$, and let $Q$ denote the set of all queries that $A$ asked its encryption oracle. The output of the experiment is 1 if and only if (1) $m \neq \bot$ and (2) $m \notin Q$. 
Authenticated Encryption

Definition: A private-key encryption scheme $\Pi$ is unforgeable if for all ppt adversaries $A$, there is a negligible function $neg$ such that:

$$\Pr[EncForge_{A,\Pi}(n) = 1] \leq neg(n).$$

Definition: A private-key encryption scheme is an authenticated encryption scheme if it is CCA-secure and unforgeable.
Generic Constructions
Encrypt-and-authenticate

Encryption and message authentication are computed independently in parallel.

\[ c \leftarrow Enc_{k_E}(m) \quad t \leftarrow Mac_{k_M}(m) \]

\[ \langle c, t \rangle \]

Is this secure? NO!
Authenticate-then-encrypt

Here a MAC tag $t$ is first computed, and then the message and tag are encrypted together.

\[ t \leftarrow Mac_{k_M}(m) \quad c \leftarrow Enc_{k_E}(m || t) \]

$c$ is sent

Is this secure? NO! Encryption scheme may not be CCA-secure.
Encrypt-then-authenticate

The message $m$ is first encrypted and then a MAC tag is computed over the result

$$c \leftarrow Enc_{k_E}(m) \quad t \leftarrow Mac_{k_M}(c) \quad \langle c, t \rangle$$

Is this secure? YES! As long as the MAC is strongly secure.