6.3. MAC на основе хэш-функции

Другим способом обеспечения целостности является использование хэш-функции. Хэш-код присоединяется к сообщению в тот момент, когда известно, что сообщение корректно. Получатель проверяет целостность сообщения вычислением хэш-кода полученного сообщения и сравнением его с полученным хэш-кодом, который должен быть передан безопасным способом. Одним из таких безопасных способов может быть шифрование хэш-кода закрытым ключом отправителя, т.е. создание подписи. Возможно также шифрование полученного хэш-кода алгоритмом симметричного шифрования, если отправитель и получатель имеют общий ключ симметричного шифрования.

Еще один вариант использования хэш-функции для получения *MAC* состоит в том, чтобы определенным образом добавить секретное значение к сообщению, которое подается на вход хэш-функции. Такой алгоритм носит название *HMAC*, и он описан в RFC 2104.

При разработке алгоритма HMAC преследовались следующие цели:

- возможность использовать без модификаций уже имеющиеся хэш-функции;
- возможность легкой замены встроенных хэш-функций на более быстрые или более стойкие;
- сохранение скорости работы алгоритма, близкой к скорости работы соответствующей хэш-функции;
- возможность применения ключей и простота работы с ними.

В алгоритме *HMAC* хэш-функция представляет собой "черный ящик". Это, во-первых, позволяет использовать существующие реализации хэш-функций, а во-вторых, обеспечивает легкую замену существующей хэш-функции на новую.

Введем следующие обозначения:

H - встроенная хэш-функция.

b - длина блока используемой хэш-функции.

n - длина хэш-кода.

K - секретный ключ. К этому ключу слева добавляют нули, чтобы получить b-битовый ключ K^+.

Вводится два вспомогательных значения:

Ipad - значение '00110110', повторенное b/8 раз.

Opad - значение '01011010', повторенное b/8 раз.

Далее *HMAC* вычисляется следующим образом:

$$HMAC = H((K^+ \oplus Opad) \,||\, H((K^+ \oplus Ipad) \,||\, M))$$