COMBINING MULTIPLE TAGGING ALGORITHMS FOR KEYED MESSAGE AUTHENTICATION

DWAYNE LITZENBERGER

Abstract. We propose a method for combining multiple independent tagging (MAC) algorithms to produce a tag whose length is independent of the number of algorithms used, and whose resistance to forgery is at least as strong as the strongest tagging algorithm used.

1. Introduction

Recent cryptanalysis of the MD4/MD5/SHA/RIPEMD family of one-way hash functions have led security experts to call for the migration away from these functions, including SHA-1. However, it is not clear to system designers which function they should migrate *to* for new designs that are expected to be used for more than a few years. While standard and custom microprocessors have become faster and more capable over the years in accordance with Moore's law, limits on the space available for message authentication have grown much less rapidly.[1] Under these circumstances, system designers may be willing to increase computation in order to increase the security of a message authentication scheme, provided that there is little or no increase in the size of individual messages.

We propose a generic scheme for message authentication using multiple keyed MAC algorithms, and show that this scheme, which does not increase message overhead beyond they of a scheme using a single MAC algorithm, is at least as strong as the *strongest* MAC algorithm used.

2. Concrete Description of the Scheme

Suppose we are given four hash functions, A, B, C, and D, such that one of these functions is collision resistant, but we do not know which one. Our goal is to use these functions to construct a keyed MAC that is also collision resistant, even if the other functions are under the *complete* control of an adversary.

Let $A(k, m)$ be a keyed MAC, which takes a security parameter k and a message m and outputs a fixed-length string of length l. Let $B(\cdot, \cdot)$ be an arbitrary function under the control of an adversary, with the only constraint being that the function must output a fixed-length string of length l. Let $\oplus$ denote the bitwise exclusive-or (XOR) operation.

We propose a new function $\mathcal{T}$, which takes two security parameters, k_1 and k_2, and a message m, and outputs a fixed-length string of length l:

$$\mathcal{T}(k_1, k_2, m) = A(k_1, m) \oplus B(k_2, m)$$

Date: March 8, 2006.

[1]For example, the minimum Internet maximum transmission unit (MTU) remains 576 octets, and is typically never more than 1500 octets.