Information Processing Letters 115 (2015) 899–902

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

On the security margin of MAC striping

T. Eisenbarth [a,*], A. Meyerowitz [b,*], R. Steinwandt [b,*]

[a] Worcester Polytechnic Institute (WPI), 100 Institute Rd, Worcester, MA 01609, United States
[b] Florida Atlantic University (FAU), 777 Glades Rd, Boca Raton, FL 33431, United States

ARTICLE INFO

Article history:
Received 5 February 2015
Received in revised form 8 April 2015
Accepted 21 May 2015
Available online 27 May 2015
Communicated by Marc Fischlin

Keywords:
Cryptography
Cryptanalysis
MAC
MAC striping
Selective forgery

ABSTRACT

MAC striping intermixes a payload with its authentication tag, placing the bits used for message authentication in positions derived from a secret key. The use of MAC striping has been suggested to authenticate encrypted payloads using short tags. For an idealized MAC scheme, the probability of a selective forgery has been estimated as $\binom{\ell+m}{m}^{-1} \cdot 2^{-m}$, when utilizing MAC striping with ℓ-bit payloads and m-bit tags. We show that this estimate is too optimistic. For $m \le \ell$ and any payload, we achieve a selective forgery with probability $\ge \binom{\ell+m}{m}^{-1}$, and usually many orders of magnitude more than that.

1. Introduction

When dealing with sensor networks, e.g., in mHealth applications, it may be necessary to authenticate very short payloads. A standard solution with a Message Authentication Code (MAC) can result in drastic data overhead. Mare et al. proposed *MAC striping* as a technique to alleviate this problem to some extent [1,2]. This technique intermixes a payload with its authentication tag, placing the bits used for message authentication at random positions, determined by a secret key. For a tag with m bits and a payload with ℓ bits, a security margin of $\binom{\ell+m}{m} \cdot 2^m$ was attributed to this construction, suggesting the use of remarkably short tags.

After recalling the description of MAC striping and discussing the notion of a selective forgery in this context, we show that the security margin of MAC striping has been overestimated significantly.

2. Preliminaries

Commonly, a MAC algorithm is formalized as a triple of algorithms:

Key generation: this algorithm chooses a secret key K.
Tag generation: this algorithm receives as input a payload P and the secret key K, and outputs a tag T for P.
Verification: this algorithm receives as input the secret key K, a payload P, and a tag T and outputs valid or invalid.

This formalization is tailored for a scenario where the tag is expected to be sent separately from the payload to be verified.

2.1. MAC striping

Instead of *concatenating* payload and tag, MAC striping *intermixes* them [1,2]. For each message to be processed, fresh tag bit positions are chosen by means of a pseudo random generator function which depends, among other

[*] Corresponding authors.
E-mail addresses: teisenbarth@wpi.edu (T. Eisenbarth), meyerowi@fau.edu (A. Meyerowitz), rsteinwa@fau.edu (R. Steinwandt).

http://dx.doi.org/10.1016/j.ipl.2015.05.009
0020-0190/© 2015 Elsevier B.V. All rights reserved.

things, on a secret key.[1] For an ℓ-bit payload P and m-bit tag T, there are $\binom{\ell+m}{m}$ choices to place the tag bits in the intermixed *message*. We assume that the tag T is chosen uniformly at random in $\{0,1\}^m$ and the set S of tag bit positions is chosen uniformly at random among all $\binom{\ell+m}{m}$ size-m subsets of $\{1,\dots,\ell+m\}$. If the same payload is processed repeatedly, each time a new tag and new tag positions are selected. This is consistent with the security analysis in [2], stating

> "to forge a message of his choice, the adversary has to guess the matching MAC bits out of 2^m possibilities, and MAC-bit locations out of $\binom{\ell+m}{m}$ possible MAC-bit locations, making the probability of success $\frac{1}{2^m\binom{\ell+m}{m}}$."

The verification algorithm does not receive payload and tag as separate inputs. Instead, it receives as input a single message M of length $m+\ell$ and the (payload-independent) secret key. The original analysis overlooks the fact that a given payload P can lead to the same message for several choices S and T. An extreme example is the payload $P=0^\ell$ in combination with the tag $T=0^m$, which for every choice of S results in the same message $0^{m+\ell}$. In general, the probability $p(M|P)$ that a particular payload P results in a message M is

$$p(M|P)=\frac{c(M,P)}{2^m\binom{\ell+m}{m}}$$

where $c(M,P)$ is the number of position/tag pairs (S,T) that produce M from P. In other words, $c(M,P)$ is the number of ways to obtain P from M by deleting m bits. So $\sum_{P\in\{0,1\}^\ell} c(M,P)=\binom{\ell+m}{m}$ and $\sum_{M\in\{0,1\}^{\ell+m}} c(M,P)=\binom{\ell+m}{m}2^\ell$.

2.2. Selective forgeries

With a standard definition of a selective forgery, the adversary is asked to generate a valid payload/tag pair (P,T) for a payload of his choice. Following the usual convention from signature schemes, to be "of the adversary's choice" it suffices to commit to the payload prior to the attack. With this convention, restricting to the payload $P=0^\ell$ is legitimate. We also consider the universal scenario where $P\in\{0,1\}^\ell$ is provided to the adversary before the attack.

When using MAC striping for a given payload P, the adversary does not need to select a tag T and position set S. If he is able to generate a message M which the verification algorithm classifies as valid and where the correct payload P is recovered by the recipient, the selective forgery succeeded: the chosen payload has been accepted as authentic. Note that the requirement of [2] that the payload P is an encrypted version of the intended text is immaterial here, as the verification and adversary both interact only with the payload P.

3. Reevaluating the security margin

Fix a payload $P\in\{0,1\}^\ell$ which the adversary will attempt to have authenticated. Our adversary will only submit messages of the correct length $\ell+m$ with $c(M,P)>0$. The number of such messages turns out to be independent of P and is given in the following lemma. Diggavi et al. [3] attribute the first part of this result to Chvátal and Sankoff [4]. We provide a proof in Appendix A.

Lemma 1. *The number of binary sequences of length $\ell+m$ containing a fixed $P\in\{0,1\}^\ell$ as a subsequence is $\sum_{k=0}^{m}\binom{\ell+m}{k}$. For $1<m<\ell$, this number is greater than $\binom{\ell+m}{m}$ and less than $\frac{\ell}{\ell-m}\binom{\ell+m}{m}$.*

Hence, submitting (any of) the most likely message(s) of length $\ell+m$ containing P as a subsequence results in a successful forgery with probability *at least* $\left(\sum_{k=0}^{m}\binom{\ell+m}{k}\right)^{-1}$.

For a given payload P, there are at least m messages with $c(M,P)=1$: distribute the tag bits at the beginning and/or end of the message and make each different than the initial or terminal bit of P. Each of these messages is correct with probability $\frac{1}{2^m\binom{\ell+m}{m}}$. So one can expect that the most likely messages are accepted with probability much greater.

Example. The values $\ell=80$ and $m=16$ have been considered in [2], and the success probability for a selective forgery is estimated to be $\frac{1}{2^{16}\binom{96}{16}}\approx 2^{-75.20}$. From Lemma 1 we see that the number of 96-bit messages containing any given 80-bit payload is $\sum_{k=0}^{16}\binom{96}{k}\approx 2^{59.51}$, and the upper bound given for this sum was $\frac{80}{80-16}\binom{96}{16}\approx 2^{59.52}$. This guarantees a success probability of $\approx 2^{-59.51}$, but the adversary can do much better. Let us look at two specific payloads which we suspect give the extremes.

$P=0^{80}$: The message 0^{96} succeeds with probability 2^{-16} (note that the length of the payload is irrelevant here).

$P=(01)^{40}$: The message $(01)^{48}$ succeeds with probability $\frac{\binom{88}{8}}{2^{16}\binom{96}{16}}\approx 2^{-39.30}$. Where does the numerator come from? We must delete some 16 bits leaving the desired payload. When we choose a certain bit to delete we have no choice but to delete the next bit as well. Then we are free to keep or delete the following bit. So the deleted bits must be 8 adjacent pairs; any such choice will work. So, consider all the ways to line up eight 1×2 rectangular boxes and eighty 1×1 squares in a line, there are $\binom{88}{8}$. Now write $\{0,1\}^{48}$ in order with 2 bits in each rectangle and one in each square. The bits in the squares form the payload.

3.1. Selective forgery for arbitrary prescribed payload

Fix a payload $P=(P_1,\dots,P_\ell)\in\{0,1\}^\ell$. We give a method to find a message M with $c(M,P)\geq 2^m$ assum-

[1] In the context considered by Mare et al., a message *header* is involved as well, but it is immaterial for our analysis of MAC striping. So we omit a discussion of header information.