UMAC Security Bound from PRP-Advantage

J. Black S. Halevi H. Krawczyk T. Krovetz P. Rogaway

November 14, 2005

Introduction. UMAC [6] is a Carter-Wegman MAC [4, 8] based on the UHASH family of hash functions. A Carter-Wegman MAC uses a family of hash-functions H and a family of masking functions F to authenticate a message M using a nonce N and private key (K', K) by associating to M a tag $H_{K'}(M) \oplus F_K(N)$. The original proof for UMAC [3, 5] established that UHASH is an ε-SU family of hash functions (strongly-universal [4], to be reviewed shortly) for a suitably small value of ε.[1] This indicates, as with any Carter-Wegman MAC, that an adversary's probability to forge against UMAC does not exceed by more than ε the ability of an adversary, with comparable resources, to break the underlying family of masking functions in the sense of distinguishing it from the family of all functions with the appropriate domain and range, a security notion known as PRF-advantage.

Beginning with Shoup [7] there has been an interest in improving the analysis of the Carter-Wegman construction when the family of masking functions is a block cipher, or is based on one, with its security measured in the sense of distinguishing it from the family of all permutations (not functions) with the appropriate domain and range, a security notion known as PRP-advantage. PRP-advantage is a tighter measure of block-cipher security—it more accurately models these objects—thereby making a preferable starting point for reductions.

The purpose of this note is to enumerate the main security results about UMAC [3, 5] as extended by applying the latest results that relate the security of a Carter-Wegman MAC to the PRP-security of a block cipher that it uses [2]. Nothing in this note is difficult, but it seems worthwhile to document how to glue together the known results, and this has been requested on the CFRG mailing list.

Definitions. See [6] for all algorithm specifications. Let $\mathcal{K}$ and $\mathcal{K}'$ be finite nonempty sets, let $n \geq 128$ (one expects $n = 128$), and let $F\colon\ \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ and $F'\colon\ \mathcal{K}' \times \{0,1\}^n \to \{0,1\}^n$ be functions. For $i \in \{1,2,3,4\}$ let UMAC-32$i[F]$ be UMAC-32i based on F; let UMAC*-32$i[F', F]$ be identical except that the key L1Key $\|$ L2Key $\|$ L3Key1 $\|$ L3Key2 for UHASH and the key K for F is determined using F' and not F; and let UHASH-32$i[F]$ be UHASH-32i using a key determined by F.

We say that $H\colon\ \mathcal{K} \times \mathcal{M} \to \{0,1\}^\tau$ is ε-SU if for all distinct $M, M' \in \mathcal{M}$ and all $Y, Y' \in \{0,1\}^\tau$ we have that $\Pr_K[H_K(M) = Y] = 2^{-\tau}$ and $\Pr_K[H_K(M) = Y \mid H_K(M') = Y'] \leq \varepsilon$. Let Func$(a, b)$ denote all functions from a-bits to b-bits and let Perm(n) denote all permutations on n bits. Let $x \xleftarrow{\$} S$ denote assigning to x a value chosen uniformly from the finite set S. Let A be an algorithm with an oracle and let $\Pr[A^{\mathcal{O}} \Rightarrow 1]$ be the probability that it outputs 1. Define $\mathrm{Adv}_F^{\mathrm{prp}}(A) = \Pr[K \xleftarrow{\$} \mathcal{K} :\ A^{F_K} \Rightarrow 1] - \Pr[\pi \xleftarrow{\$} \mathrm{Perm}(n) :\ A^{\pi} \Rightarrow 1]$ and $\mathrm{Adv}_F^{\mathrm{prf}}(A) = \Pr[K \xleftarrow{\$} \mathcal{K} :\ A^{F_K} \Rightarrow 1] - \Pr[\rho \xleftarrow{\$} \mathrm{Func}(n, n) :\ A^{\rho} \Rightarrow 1]$. For MAC: $\mathcal{K} \times \mathcal{N} \times \mathcal{M} \to \{0,1\}^\tau$ let $\mathrm{Adv}_{\mathrm{MAC}}^{\mathrm{mac}}(A) = \Pr[K \xleftarrow{\$} \mathcal{K} :\ A^{\mathrm{MAC}_K(\cdot,\cdot)}$ forges] where A is said to forge if it asks a sequence of queries $(M_1, N_1), \ldots, (M_q, N_q)$ with distinct N_i-values and then outputs an (M, N, T) where $T = \mathrm{MAC}_K(M, N)$ and A never asked a query (M, N). Adversaries are assumed to never repeat a query. For each advantage measure xxx let $\mathrm{Adv}_{\Pi}^{\mathrm{xxx}}(t, q) = \max_A\{\mathrm{Adv}_{\Pi}^{\mathrm{xxx}}(A)\}$ where A's running time plus encoding size is at most t (in some fixed model of computation) and A asks at most q oracle queries. We say that (t', q') is comparable to (t, q) if $t' \leq ct \lg t$ and $q' \leq q + c'$ for constants c and c' determined from the reduction in question, details of the model of computation, and the parameter n. (We adopt this shorthand because stating more precise resource-bound comparisons gives rise to more obscure-looking statements.)

Results. Fix $n \geq 128$ and $i \in \{1,2,3,4\}$. Below we let A be an adversary, $t, q, \tau \geq 1$ numbers, and $\mathcal{K}, \mathcal{M}$ nonempty sets, the former finite. After the following sequence of results we briefly describe each step.

[1] The value ε actually depends on the length of the tag generated, with 32, 64, 96, and 128 bits permitted by [6].

Lemma 1 $\mathrm{UHASH\text{-}32}i[\mathrm{Func}(n,n)]$ *is* ε-SU *where* $\varepsilon = (.562502^i \cdot 2^{-30\,i})$.

Lemma 2 *Let* $H\colon \mathcal{K} \times \mathcal{M} \to \{0,1\}^\tau$ *be* ε*-SU, let* $f \in \mathrm{Func}(n,\tau)$*, and let* $\mathrm{MAC}_{K,f}(M,N) = H_K(M) \oplus f(N)$ *be the Carter-Wegman MAC based on* H *and* $\mathrm{Func}(n,\tau)$*. Then* $\mathrm{Adv}^{\mathrm{mac}}_{\mathrm{MAC}}(A) \le \varepsilon$.

Lemma 3 *Let* $\Pi = \mathrm{UMAC}^*\text{-}32i[\mathrm{Func}(n,n), \mathrm{Func}(n,n)]$*. Then* $\mathrm{Adv}^{\mathrm{mac}}_{\Pi}(A) \le (.562502^i \cdot 2^{-30\,i})$.

Theorem 1 *Let* $E\colon \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ *and let* $\Pi = \mathrm{UMAC\text{-}32}i[E]$*. Then* $\mathrm{Adv}^{\mathrm{mac}}_{\Pi}(t,q) \le 2^{-30\,i} + 2\mathrm{Adv}^{\mathrm{prf}}_{E}(t',q')$ *where* (t',q') *is comparable to* (t,q)*. Also* $\mathrm{Adv}^{\mathrm{mac}}_{\Pi}(t,q) \le 2^{-30\,i} + 2\mathrm{Adv}^{\mathrm{prp}}_{E}(t',q') + (q')^2/2^n$.

Lemma 4 $\Pr[\pi \xleftarrow{\$} \mathrm{Perm}(n) : A^\pi \Rightarrow 1] \le c \cdot \Pr[\rho \xleftarrow{\$} \mathrm{Func}(n,n) : A^\rho \Rightarrow 1]$ *where* $c = 1.7$ *if* A *asks* $q \le 2^{64}$ *queries and* $n \ge 128$*, and where* $c = 1.01$ *if* A *asks* $q \le 2^{10}$ *queries and* $n \ge 128$.

Lemma 5 *Let* $\Pi = \mathrm{UMAC}^*\text{-}32i[\mathrm{Func}(n,n), \mathrm{Perm}(n)]$, $q < 2^{64}$*. Then* $\mathrm{Adv}^{\mathrm{mac}}_{\Pi}(t,q) \le 1.7\,(.562502^i \cdot 2^{-30\,i})$.

Lemma 6 *Let* $E\colon \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ *and* $\Pi = \mathrm{UMAC}^*\text{-}32i[\mathrm{Func}(n,n), E]$*. Assume* $q < 2^{64}$*. Then* $\mathrm{Adv}^{\mathrm{mac}}_{\Pi}(t,q) \le 1.7\,(.562502^i \cdot 2^{-30\,i}) + \mathrm{Adv}^{\mathrm{prp}}_{E}(t',q')$ *where* (t',q') *is comparable to* (t,q).

Lemma 7 *Let* $E\colon \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ *and* $\Pi = \mathrm{UMAC}^*\text{-}32i[\mathrm{Perm}(n), E]$*. Assume* $q < 2^{64}$*. Then* $\mathrm{Adv}^{\mathrm{mac}}_{\Pi}(t,q) \le 1.72\,(.562502^i \cdot 2^{-30\,i}) + 1.01\,\mathrm{Adv}^{\mathrm{prp}}_{E}(t',q')$ *where* (t',q') *is comparable to* (t,q).

Lemma 8 *Let* $E\colon \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ *and* $\Pi = \mathrm{UMAC\text{-}32}i[E]$*. Assume* $q < 2^{64}$*. Then* $\mathrm{Adv}^{\mathrm{mac}}_{\Pi}(t,q) \le 1.72\,(.562502^i \cdot 2^{-30\,i}) + 1.01\,\mathrm{Adv}^{\mathrm{prp}}_{E}(t',q') + \mathrm{Adv}^{\mathrm{prp}}_{E}(t'',q'')$ *where* (t',q') *and* (t'',q'') *are comparable to* (t,q).

Theorem 2 *Let* $E\colon \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ *and let* $\Pi = \mathrm{UMAC\text{-}32}i[E]$*. Assume* $q < 2^{64}$*. Then* $\mathrm{Adv}^{\mathrm{mac}}_{\Pi}(t,q) \le 2^{-30\,i} + 3\,\mathrm{Adv}^{\mathrm{prp}}_{E}(t',q')$ *where* (t',q') *is comparable to* (t,q).

Explanation. Lemma 1 is the central result, taken from [5]. The actual bound is $\varepsilon = (2^{-31} + 2^{-34} + 2^{-49})^i$; it is stated here in a more convenient form. The different addends arise from the different layers of hashing. This bound is achieved by substituting $\mathsf{NH}[32]$ instead of $\mathsf{NHS}[16]$ into the proof of [5, Theorem 6.4.5]. Lemma 2 is the standard result giving the security of the Carter-Wegman MAC [4, 8]. Lemma 3 is the immediate combining of Lemmas 1 and 2. Theorem 1 is obtained from Lemma 3 by a standard argument: replace the second $\mathrm{Func}(n,n)$ in the statement of Lemma 3 by $E\colon \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$, contributing a term of $\mathrm{Adv}^{\mathrm{prf}}_{E}(t',q+1)$; then replace the first $\mathrm{Func}(n,n)$ by E, contributing a term of $\mathrm{Adv}^{\mathrm{prf}}_{E}(t'',92)$; and then weaken the statement by dropping the $.562502^i$. The statement following the "also" is obtained by applying the PRP/PRF switching lemma (where, to be concrete, $q' = \max\{q+1, 92\}$).

If one assumes $\mathrm{Adv}^{\mathrm{prf}}_{E}(t',q')$ to be sufficiently small then the first bound in Theorem 1 suffices and further refinement is not needed. If one instead wants to base the analysis solely on the quality of E as a PRP then one needs the second bound in Theorem 1. In that case a better bound can be obtained using recent results from [2], which effectively allow one to disregard the $(q')^2/2^n$ term as long as the number of queries is not too large. Lemma 4 is adapted from [2], where the more general statement is given that for an adversary A that makes q queries, $\Pr[A^\pi \Rightarrow 1] \le c \cdot \Pr[A^\rho \Rightarrow 1]$ where $c = (1 - q/2^n)^{-q/2}$. Use $1 + x \approx e^x$ (for $x \approx 0$) to get a feel for this, while somewhat more playing with inequalities and plugging in the numbers is needed to derive Lemma 4 from [2, Theorem 2.3]. Lemma 5 is obtained by combining Lemmas 3 and 4. An adversary A attacking the MAC can, with one additional query, be regarded as an adversary that outputs 1 any time the former MAC-attack is successful. Apply Lemma 4 to this adversary and re-interpret the result in terms of attacking the MAC. Lemma 6 is obtained from Lemma 5 in the usual manner of information-theoretic to complexity-theoretic conversion. Given a MAC-attacking adversary A one constructs a distinguisher B that distinguishes E_K (for a random key K) from π (a random permutation from $\mathrm{Perm}(n)$) in the natural way. Lemma 7 is obtained by combining Lemmas 4 and 6, much as before except that now the maximum number of queries is $92 < 2^{10}$, which is how many queries it takes to produce the UHASH internal key and the key for the function family E. Lemma 8 is obtained from Lemma 7 in the standard way that one passes from information-theoretic to complexity theoretic results in provable-security cryptography. Theorem 2 is the final result, a simple rewriting of Lemma 8 to improve readability and simplify the constants.

This note analyzes the case of a single forgery attempt by an attacker. It is a straightforward exercise to show that over q_{v} forgery attempts the probability that an attacker gets at least one right is no more than $q_{\mathrm{v}}\, 2^{-30\,i} + 3\,\mathrm{Adv}^{\mathrm{prp}}_{E}(t',q')$.

References

[1] M. Bellare, O. Goldreich, A. Mityagin. The power of verification queries in message authentication and authenticated encryption. Cryptology ePrint Archive, Report 2004/309, 2004.

[2] D. Bernstein. Stronger security bounds for permutations. Unpublished manuscript, 2005, available from the author's web page.

[3] J. Black, S. Halevi, H. Krawczyk, T. Krovetz, and P. Rogaway. UMAC: Fast and provably secure message authentication. *CRYPTO '99*, pp. 216–233, Springer, 1999.

[4] L. Carter and M. Wegman. Universal classes of hash functions. *J. of Computer and System Sciences*, vol. 18, pp. 143–154, 1979.

[5] T. Krovetz. Software-optimized universal hashing and message authentication. MI Dissertation Services, 2000.

[6] T. Krovetz, editor, with J. Black, S. Halevi, H. Krawczy, and P. Rogaway. UMAC: Message authentication using universal hashing. Internet Draft draft-krovetz-umac-07.txt, November 2005.

[7] V. Shoup. On fast and provably secure message authentication based on universal hashing. *CRYPTO '96*, pp. 313–328, 1996.

[8] M. Wegman and L. Carter. New hash functions and their use in authentication and set equality. *J. of Computer and System Sciences*, vol. 22, pp. 265–279, 1981.