12.7 Literaturverzeichnis

1. ANSI X9.17-1985. American National Standard X9.17: Financial Institution Key Management, 1985.
2. Mihir Bellare, Ran Canetti, and Hugo Krawczyk. Keying Hash Functions for Message Authentication. In *CRYPTO '96: Proceedings of the 16th Annual International Cryptology Conference, Advances in Cryptology*, pages 1–15. Springer, 1996.
3. Mihir Bellare, Ran Canetti, and Hugo Krawczyk. Message Authentication using Hash Functions—The HMAC Construction. *CRYPTOBYTES*, 2, 1996.
4. J. Black, S. Halevi, H. Krawczyk, T. Krovetz, and P. Rogaway. UMAC: Fast and secure message authentication. In *CRYPTO '99: Proceedings of the 19th Annual International Cryptology Conference, Advances in Cryptology*, volume 99, pages 216–233. Springer, 1999.
5. C. M. Campbell. Design and specification of cryptographic capabilities. *NBS Special Publication 500-27: Computer Security and the Data Encryption Standard, U.S. Department of Commerce, National Bureau of Standards*, pages 54–66, 1977.
6. J.L. Carter and M.N. Wegman. New hash functions and their use in authentication and set equality. *Journal of Computer and System Sciences*, 22(3):265–277, 1981.
7. Morris Dworkin. Recommendation for Block Cipher Modes of Operation: The CCM Mode for Authentication and Confidentiality, May 2004. `http://csrc.nist.gov/publications/nistpubs/800-38C/SP800-38C_updated-July20_2007.pdf`.
8. Morris Dworkin. Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, NIST Special Publication 800-38D, May 2005. `http://csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf`.
9. Morris Dworkin. Recommendation for Block Cipher Modes of Operation: Galois Counter Mode (GCM) and GMAC, NIST Special Publication 800-38D, November 2007. `http://csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf`.
10. S. Halevi and H. Krawczyk. MMH: message authentication in software in the Gbit/second rates. In *Proceedings of the 4th Workshop on Fast Software Encryption*, volume 1267, pages 172–189. Springer, 1997.
11. D. McGrew and J. Viega. RFC 4543: The Use of Galois Message Authentication Code (GMAC) in IPsec ESP and AH. Technical report, Corporation for National Research Initiatives, Internet Engineering Task Force, Network Working Group, May 2006. Available at `http://rfc.net/rfc4543.html`.
12. J.H. Song, R. Poovendran, J. Lee, and T. Iwata. RFC 4493: The AES-CMAC Algorithm. Technical report, Corporation for National Research Initiatives, Internet Engineering Task Force, Network Working Group, June 2006. Available at `http://rfc.net/rfc4493.html`.
13. NIST Special Publication SP800-38D: Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, November 2007. Available at `http://csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf`.
14. D. Whiting, R. Housley, and N. Ferguson. RFC 3610: Counter with CBC-MAC (CCM). Technical report, Corporation for National Research Initiatives, Internet Engineering Task Force, Network Working Group, September 2003.

Aufgaben

12.1. In diesem Kapitel wurde gezeigt, dass MACs zur Authentisierung von Nachrichten verwendet werden können. In dieser Aufgabe betrachten wir den Unterschied zwischen einem Protokoll mit einem MAC und einem mit einer digitalen Signatur. Der Sender führt die folgende Operation aus:

1. Protokoll A:
$$y = e_{k_1}[x||h(k_2||x)],$$
wobei x die Nachricht ist, $h()$ eine Hashfunktion (beispielsweise SHA-1), e eine symmetrische Chiffre und k_1, k_2 geheime Schlüssel, die nur dem Sender und Empfänger bekannt sind. Das Symbol „||" steht für die Verkettung von Werten.
2. Protokoll B:
$$y = e_k[x||sig_{k_{pr}}(h(x))]$$

Beschreiben Sie jeden Schritt, den der Empfänger durchführt, nachdem der Wert y bei ihm eingeht. Empfehlenswert, aber optional ist es, ein Blockdiagramm für den Prozess auf der Empfängerseite zu zeichnen.

12.2. Um Angriffe, die auf dem Geburtstagsparadoxon basieren, abzuwenden, müssen Hashfunktionen eine hinreichend lange Ausgangswortbreite aufweisen, z. B. 160 Bit. Warum sind für MACs wesentlich kürzere Ausgangswörter ausreichend, z. B. 80 Bit?

Gehen Sie von folgender Annahme aus: Eine Nachricht wird unverschlüsselt mit ihrem entsprechendem MAC versandt: $(x, MAC_k(x))$. Beschreiben Sie genau, was ein Angreifer Oskar tun muss.

12.3. Wir untersuchen zwei Möglichkeiten zum Integritätsschutz mit Verschlüsselung.

1. Wir betrachten ein Schema, bei dem Verschlüsselung und Integritätsschutz zusammen geboten werden, indem das Chiffrat in der folgenden Weise berechnet wird:
$$c = e_k(x||h(x))$$
Hierbei ist $h()$ eine Hashfunktion. Dieses Verfahren ist angreifbar, wenn für die Verschlüsselung eine Stromchiffre verwendet wird und der Angreifer den gesamten Klartext kennt. Beschreiben Sie im Detail, wie ein Angreifer den Klartext x mit einem beliebigen anderen Klartext x' ersetzen kann und hierfür ein c' berechnen kann, so dass der Empfänger die Nachricht als gültig verifiziert. Wir nehmen an, dass x und x' gleich lang sind. Kann dieser Angriff auch durchgeführt werden, wenn mit einem One-Time-Pad verschlüsselt wird?
2. Ist der Angriff auch möglich, wenn eine Hashfunktion mit Schlüssel, beispielsweise ein MAC, verwendet wird?
$$c = e_{k_1}(x||MAC_{k_2}(x))$$
Wir nehmen nach wie vor an, dass für die Verschlüsselung eine Stromchiffre zum Einsatz kommt.

12.4. In dieser Aufgabe werden die Probleme einer effizienten MAC-Konstruktion diskutiert.

1. Die Nachricht x, die authentisiert werden soll, besteht aus z unabhängigen Blöcken, wobei $X = x_1||x_2||\ldots||x_z$ und jedes x_i aus acht Bits besteht. Die Eingangswerte werden iterativ in der Kompressionsfunktion verarbeitet:

$$c_i = h(c_{i-1}, x_i) = c_{i-1} \oplus x_i$$

Am Ende wird der MAC-Wert wie folgt berechnet:

$$MAC_k(X) = c_z + k \bmod 2^8,$$

wobei k ein gemeinsamer 64 Bit langer Schlüssel ist. Beschreiben Sie, wie der effektiv wirksame Teil des Schlüssels mit nur einer einzigen Nachricht x berechnet werden kann.

2. Führen Sie den Angriff mit dem folgenden Parametern durch und bestimmen Sie den Schlüssel k:

$$\begin{aligned} X &= \texttt{HELLO ALICE!} \\ c_0 &= 11111111_2 \\ MAC_k(X) &= 10011101_2 \end{aligned}$$

3. Was ist die effektive Länge des Schlüssels k?
4. Obwohl diese MAC-Konstruktion zwei verschiedene Operationen ($[\oplus, 2^8]$ und $[+, 2^8]$) verwendet, weist sie doch erhebliche Schwachstellen auf. Worauf basieren diese Schwachstellen? Worauf sollte man achten, wenn man ein Kryptosystem entwirft? Diese essentielle Eigenschaft ist auch für Blockchiffren und Hashfunktionen wichtig.

12.5. Prinzipiell können MACs auch mit Kollisions-Attacken angegriffen werden. Wir diskutieren diese Fragestellung im Folgenden.

1. Wir nehmen an, Oskar hat eine Kollision zweier Nachrichten gefunden:

$$\mathrm{MAC}_k(x_1) = \mathrm{MAC}_k(x_2)$$

Zeigen Sie ein einfaches Protokoll, mit der diese Attacke ausgenutzt werden kann.

2. Prinzipiell ist das Geburtstagsparadoxon auch hier anwendbar. In der Praxis ist es dennoch viel schwieriger, das Geburtstagsparadoxon auf MACs anzuwenden, als auf Hashfunktionen. Aufgrund dieser Beobachtung können wir die folgende Frage beantworten: Welche Sicherheit bietet ein MAC mit 80-Bit-Ausgangswert verglichen mit einer Hashfunktionen mit 80-Bit-Ausgangswert?