

# SLC: Efficient Authenticated Encryption for Short Packets

Ammar Alkassar  
a.alkassar@sirrix.com  
Sirrix AG (DE)

Elena Andreeva  
elena.andreeva@esat.kuleuven.be  
KU Leuven (BE)

Helger Lipmaa  
lipmaa@ut.ee  
Cybernetica AS and University of Tartu (EE)

**Abstract:** We present a new, provably secure, self-synchronizing authenticated encryption mode of operation, SLC, with the ability to re-synchronize after loss of transmission units of sub-block size, enabling it to efficiently handle short packets. The new scheme uses two components, self-synchronizing MAC and self-synchronizing encryption scheme, each of which is individually interesting. The SLC mode, as well as both its components, are highly parallelizable and efficient in fault-tolerant applications.

## 1 Introduction

Most current protocols and applications require mechanisms for protecting both privacy and authenticity of data. Subsequently, to ensure both security goals, the employment of either both encryption and authentication mechanisms, known as *generic composition*, or a single *authenticated encryption* (AE) mechanism, is needed. Earlier versions of protocols like SSL, SSH and WEP 802.11 version of IPsec integrated encryption and authentication schemes without carefully examining the security of the resulting generic composition, which led to security flaws and necessitated a comprehensive security analysis [BN01] of these compositions. Moreover, new AE schemes with desirable properties like: *pipelining* and *parallelizability*; *single key* usage; *reduced number of block cipher calls*; *minimal delay*; *on-line processing*; were proposed. Such characteristics are in contrast to the common generic block cipher based compositions and thus, efficient AE schemes like GCM [MV04], which is the basis for secure processing in security standards such as IEEE 802.1AE and an extension to the IETF IPsec standard, are currently considered for standardization.

However, encryption and authentication in established circuit-based networks like ISDN, GSM, SDH/SONET as well as packet-oriented networks with long packets face the problem of resolving the problem of resynchronization of the data stream when errors in transmission occur. Especially applications with natural fault-tolerance like voice and video usually refrain from data link control protocols and thus, need some kind of inband resynchronization.

None of the standard AE modes deal with resynchronization of the data stream when

errors in transmission occur. Such errors are: *bit errors*, which occur if one or a few bits are flipped (their number remains unchanged), and *slips* when bits are inserted or lost (frame-errors).

Mechanisms that overcome slips are called *self-synchronizing*, in the sense that a slip in the ciphertext leads only to a small amount of incorrect plaintext. To minimize the rejection of unverified data due to transmission errors, either expensive resynchronization protocols are wrapped around the used scheme, or authenticated encryption is applied over messages of short length (in terms of bytes). In the latter case, any error within a message would cause it to be dismissed and the data processing would be resynchronized for the next message. But this technique does not comprise the important efficiency issue that the processing of the message involves the full AE scheme invocation for every message anew. This, for example in the case of the otherwise efficient GCM mode, increases the load of processing with the accumulation of multiple short messages (GCM, similarly to other AE modes, makes extra block cipher calls).

In this paper, we sketch a self-synchronizing, single key, two-pass AE scheme SLC, that recovers efficiently from slip errors of a sub-block size (stream oriented). The new mode, SLC, is similar to the GCM mode and includes similar properties like parallelizability, fast software and hardware performance, and on-line processing. Compared to GCM, SLC adds the ability to resynchronize after transmission errors by inheriting the cheap resynchronization technique, exploited by encryption schemes like OCFB [AGPS01] and SCFB [JR99]. Moreover, we distinguish between two types of SLC variants in correspondence with the network characteristics.

In several applications that require a self-synchronizing AE scheme: circuit-based networks like ISDN, GSM or SDH/SONET, message expansion is not allowed. Additional data in these networks can be transmitted out-of-band (e.g. using D-Channel in ISDN or by voice compression for making additional space available). For such applications, we propose an *out-of-band* SLC scheme, where the authentication tags are transmitted separately from the ciphertext over the out-of-band channel.

There exist other applications, e.g. Voice-over-IP applications that allow the in-band transmissions of the tags together with the message (packet, e.g. IP) over the same channel. The packets are delivered over an error-prone channel and in order to keep synchronization, the header keeps some information that is used together with a stateful AE scheme. In our solution, the *in-band* SLC scheme sends every packet together with its tag and since the header size should be minimized, we do not add any extra information except the tag itself.

Compared to other AE schemes, SLC is especially very effective whenever short packets are required. E.g., in the VoIP media stream protocol RTP, data packets of 20-60 bytes are transmitted. Current AE schemes like GCM are applied for every packet independently, causing a significant overhead with every packet of up to 80 percent. And, making a single run of these AE schemes over multiple packets is not an option, because of their lack in resynchronization: if a single ip-packet is lost, the subsequent transmission will be lost, too.

SLC makes use of new underlying primitives, SSWC (self-synchronizing Wegman-Carter

MAC) and SCTR (self-synchronizing counter scheme), that are also defined in this paper.

## 2 Related work

The AE schemes that introduce provably secure and efficient authenticated encryption are generally divided into two major classes. The first, *conventional* AE schemes, internally make two passes over the data with a single key. The traditional CCM [WHF02] and EAX [BRW04] schemes use the CTR encryption mode and a CBC-MAC and OMAC (CBC-MAC variant) resp., while the more recent CWC [KVV04] and GCM [MV04] rely upon universal hash-based authentication [CW79]. Recent conventional designs are the CCFB AE and CCFB+H AE with associated data schemes [Luc05]. The *unconventional* schemes that process the data in a single run are: IAPM [Jut01], XCBC [GD01], and OCB [RBB03]. They introduce high efficiency, but are covered by patents. Fact is, that conventional schemes like GCM are competing efficiently with the unconventional designs (e.g. OCB) in software and hardware and are thus preferred at present.

To overcome transmission error problems that AE should deal with, our scheme, SLC, benefits from simple resynchronization techniques that use statistical properties of the ciphertext, e.g. the optimized self-synchronizing encryption mode OCFB [AGPS01]. OCFB takes advantage of the cipher-feedback encryption mode of operation (CFB) and uses recognition of a pattern in the ciphertext to re-synchronize the states of the transmitter's and receiver's key-stream generators. A general treatment of this and a similar encryption scheme, SCFB [JR99], can be found in [Hey03].

## 3 Preliminaries

**Notation:** Let  $(\mathcal{G}, +)$  be an additive commutative finite group,  $\mathcal{G} = GF(2^\beta)$ , with  $\beta$ -bit elements for some block length  $\beta > 0$ .  $\lambda$  denotes the empty string.  $|M|$  is the length and  $M[i]$  the  $i$ th bit of the bit string  $M$ .  $M[i..j]$  denotes the part of  $M$  that starts at  $M[i]$  and ends at  $M[j]$ ,  $i \leq j$ .  $M[\text{first } i \text{ bits}]$  are the first  $i$  bits of  $M$  (if  $|M| \geq i$ ), and  $M0^{i-|M|}$  (append  $0^s$ ), otherwise.  $\text{encode}_\beta(t)$  encodes  $t$  into a  $\beta$ -bit binary string (MSB first, LSB last) for  $t \in [1..2^\beta]$ .  $\parallel$  is the concatenation operation and the  $\text{match}(\cdot, \cdot)$  function compares two bit strings and returns 1, if they are equal, else 0.  $\ell$  is the length of a transmission unit (e.g.,  $\ell = 8$ ). As in [AGPS01], we use  $\ell$ -bit shift registers of  $v = \beta/\ell$  positions  $\text{sr}[1], \text{sr}[2], \dots, \text{sr}[v]$ . With  $r \leftarrow \text{sr}$  and  $\text{sr} \leftarrow r$  we denote reading, resp., writing the entire register and with  $\text{sr} \ll m$  and  $m \ll \text{sr}$  the shifting of  $\ell$  into, resp. out of the register.

**Block cipher** is a triple (Gen, Enc, Dec) of efficient algorithms. The key  $k$  is generated uniformly at random by Gen,  $k \leftarrow \mathcal{K}$  and  $\mathcal{K} = \{0, 1\}^\kappa$  for  $\kappa > 0$ .  $\text{Enc}_k : \{0, 1\}^\beta \rightarrow \{0, 1\}^\beta$  encrypts a plaintext  $M$  to a ciphertext  $C = \text{Enc}_k(M)$ .  $M = \text{Dec}_k(C)$  using the same valid key  $k$  and  $C$ . Block ciphers are modeled as pseudo-random permutations (PRP) families. Since we use block cipher modes with no decryption operation, we model the block cipher as pseudo-random function (PRF) family.

**Hash function family**  $\mathcal{H} : \mathcal{K} \times \mathcal{M} \rightarrow \mathcal{G}$  is  $\varepsilon$ -*almost xor universal* (AXU) [Kra94], if for any pair  $M_1 \neq M_2$  in  $\mathcal{M}$ , any  $\Delta \in \mathcal{G}$ , and uniform random key  $k \leftarrow \mathcal{K}$ , the probability, taken over all keys, that  $\mathcal{H}_k(M_1) \oplus \mathcal{H}_k(M_2) = \Delta$  is no more than  $\varepsilon$ , where  $\mathcal{M}$  and  $\mathcal{K}$  are finite sets. One particularly efficient  $\varepsilon$ -AXU family, PolyHash, is defined in [MV04].

**Message authentication codes (MAC)** (nonce-based) are triples of efficient algorithms (GenA, Tag, Ver). The GenA returns a uniform random key  $k \leftarrow \mathcal{K}$ .  $\text{Tag}_k : \mathcal{N} \times \mathcal{M} \rightarrow \mathcal{T}$  generates  $T = \text{Tag}_k(N, M)$ . The verification algorithm returns  $\text{Ver}_k(N, M, T) \in \{\text{accept}, \text{reject}\}$ , such that  $\text{Ver}_k(N, M, \text{Tag}_k(M)) = \text{accept}$  for all valid inputs. We say that the MAC is secure if it is existentially unforgeable under chosen message attack.

In our scheme we adopt the Wegman-Carter’s MAC [WC81, Kra94], WC. For  $\mathcal{K} = \{0, 1\}^\kappa \times \mathcal{G}$ , and  $\mathcal{M} = \{0, 1\}^{\leq m^l}$ , let  $\mathcal{P} : \mathcal{K} \times \mathcal{G} \rightarrow \mathcal{G}$  be a family of PRPs and  $\mathcal{H} : \mathcal{K} \times \mathcal{M} \rightarrow \mathcal{G}$  be  $\varepsilon$ -AXU family of functions and the key pair  $(\text{kf}, \text{kh}) \in \mathcal{K}$ . Then, WC produces  $\text{Tag}_k(N, M) = \mathcal{P}_{\text{kf}}(N) \oplus \mathcal{H}_{\text{kh}}(M)$ , where  $N \in \mathcal{N}$  identifies  $M$  uniquely. The security of the WC is proven in [Sho96] and an improved bound is analyzed in [Ber05].

**Modes of Operation** (nonce-based) are encryption schemes, SE, used when the length of a message exceeds the block length, and that take  $k$ ,  $N$  and  $M$  to produce  $C$ . In decryption, they should return the input message  $M$  on the same valid  $k$ ,  $N$  and  $C$ . The security of SE is defined in the sense of LOR (left or right), ROR (real or random), FTG (find then guess) indistinguishability or semantic security notions [BDJR97] under chosen plaintext or ciphertext attacks (CPA and CCA resp.).

## 4 New self-synchronizing schemes

### Self-synchronizing Wegman-Carter

To build SSWC, we apply the Wegman-Carter methodology to compose a strong MAC from  $\varepsilon$ -AXU hash function and PRF families. Thus, we first fix an  $\varepsilon$ -AXU hash function family (we choose PolyHash) and a PRP family  $\mathcal{P}$ . On top of the WC construction, we additionally fix a statistically chosen pattern in the ciphertext. In practice, there is no special requirement for the pattern and is chosen as system parameter. This is possible due to the assumption that the ciphertext has a uniform distribution. SSWC calculates WCs over subsequent message parts while parsing  $M$  in  $\ell$ -bit units starting from bit  $i$ , and ending at the first occurrence of pattern (position  $i'$ ), or at the message end. This operation is denoted by “Pattern-parse  $M[i \dots i']$ ” and allows implementation with shift registers [AGPS01]. The pattern is included in the resulting bit-string. We then transmit  $M[i \dots i']$  with its tag and continue from the bit  $i' + 1$  (with an increased nonce). For  $\text{WC} = (\widehat{\text{GenA}}, \widehat{\text{Tag}}, \widehat{\text{Ver}})$ , the general composition, SSWC, is defined by the next function  $\text{Tag}_k(N, M)$ :

1. Set  $i \leftarrow 1$  and  $j \leftarrow 1$ ;
2. Repeat till the end of  $M$ :
  - (a) Pattern-parse  $M[i \dots i']$ ;
  - (b) Set  $T_j \leftarrow \widehat{\text{Tag}}_{k, \mathcal{P}_k(0)}(N + j - 1), M[i \dots i']$ ,  
 $j \leftarrow j + 1, i \leftarrow i' + 1$ ;
3. Return  $T_1 \parallel \dots \parallel T_{j-1}$ .

SSWC can process long streams and the length of  $M$  need not be known until its end (on-line).

### Self-Synchronizing Counter Mode of Operation

We fix a new, self-synchronizing encryption scheme, based on the CTR scheme. As with SSWC, we add the pattern-matching mechanism to obtain the required functionality. More precisely, we apply a PRP family  $\mathcal{P}$  to a counter with an initial value  $N$ . A pattern  $p$  and an integer  $w > 0$  are fixed. Notice that in SCTR, the ciphertext is parsed on both sides for a defined  $p$ . If  $p$  is found, both counters are increased to the next  $w$ -place (counter increments stop before its repetition) and the parsed message chunks are encrypted in CTR scheme. For example, if  $w = 10$ , then the counter is rounded up to the next decimal; this enables re-synchronization after slip-errors with at most  $w$  lost blocks (similar usage of  $w$  in SSWC would make it more robust). In between the chunks, the value of the counter is increased to the next multiple of  $w$ . The new SCTR scheme is defined by  $\text{Enc}_k(N, M)$  as follows:

1. Write  $M = M_1 \parallel M_2 \dots \parallel M_{n-1} \parallel M_n$  where
2.  $|M_i| = \ell$ , pad (append)  $M_n$  with 0's if necessary;
3. Set  $v \leftarrow \beta/\ell$ ;  $j \leftarrow 0$ ;  $\text{sr1} \leftarrow 0$ ;  $s \leftarrow v$ ;
4. For  $i \in \{1, \dots, n\}$  do:
  - (a) If  $s = v$  then set  $\text{sr2} \leftarrow \mathcal{P}_k(N + j), j \leftarrow j + 1, s \leftarrow 0$ ; endif.
  - (b) Set  $O_i \ll \text{sr2}, C_i \leftarrow M_i \oplus O_i, \text{sr1} \ll C_i$ ;
  - (c) If  $\text{match}(\text{sr1}, p)$  then set  $s \leftarrow v, j \leftarrow w \cdot \lceil j/w \rceil$   
else set  $s \leftarrow s + 1$ ; endif.
5. Return  $C \leftarrow C_1 \parallel C_2 \parallel \dots \parallel C_n$ .

SCTR can resynchronize if the number of lost frames is less than  $w$ , but not if  $p$  is lost. SCTR has desirable characteristics like no error propagation (on bit errors) and high parallelizability.

### New Self-Synchronizing AE scheme SLC

1. The *out-of-band version* of SLC consists of first encrypting the message by using a self-synchronizing encryption scheme (SCTR, OCFB) to produce the ciphertext  $C$ , and then applying a self-synchronizing MAC (uses subkey  $k_{\text{kh}} \leftarrow \text{Enc}_k(N)$ ) algorithm (in our case, SSWC based on PolyHash) to the ciphertext to produce the tag  $T$ . Both, the ciphertext and the tag are returned. The out-of-band version of SLC corresponds to a general composition and is done just in the straightforward way.

2. The *in-band-version* is slightly more complicated. During encryption, we encrypt chunks (separated by  $p$ ) using a conventional encryption scheme, and accompany every chunk with its tag. The decryption operation first splits the ciphertext into chunks, according to the pattern  $p$ , then recomputes a tag over every individual chunk and *iff* it coincides

with the tag given by  $T$ , decrypts the corresponding chunk of the ciphertext and appends the result to the final plaintext. As we see, in this SLC version, the implementation of a standard MAC suffices.

**Properties of SLC:** 1. Unlike any of the previously proposed AE schemes, SLC *resynchronizes after both bit errors and slips*. Even if a tag in SLC gets slipped, the synchronization can be restored at least after the next occurrence of the synchronization pattern; 2. SLC is highly *parallelizable*, i.e., the encryption can be done by parallel block cipher engines working concurrently (with SCTR) and the authentication part can also be implemented for parallel processing (with SSWC); 3. Our scheme is *on-line*, it is able to process the stream of data as it arrives, not knowing its end (advantageous in circuit-based communication systems); 4. If used in conjunction with the counter scheme, SLC has *minimal delay*. Another possible advantage of decoupling the heavy processing from the data is the ability to be used in hybrid software-hardware systems (HSHS), allowing high-speed encryption and authentication with cheap conventional hardware.

**Security considerations:** To prove the privacy of SLC, it suffices to show the privacy of the underlying encryption mode (e.g. SCTR), because the MAC just gets the ciphertext to calculate the tags. Hence, we first prove that the used underlying encryption scheme preserves privacy under CPA. The security of SCTR under CPA is proven in the LOR indistinguishability model [BDJR97].

Initially, we model the block cipher as a PRF, but finally applying the PRF-PRP lemma [BDJR97], we conclude the CPA security of the SCTR scheme and the CPA privacy of the SLC scheme with a PRP.

**Theorem 1 (Privacy of SLC)** Let  $Enc$  be a family of permutations from  $\mathcal{G} = GF(2^\beta)$  to  $\mathcal{G} = GF(2^\beta)$ . Then for any time  $t$ , encryption queries  $q_e$  of total length  $\mu_e \leq \frac{2^\beta}{w}$ , an adversary  $A$  has distinguishing advantage

$$\mathbf{Adv}_{\text{SLC}[Enc]}^{\text{ind-cpa}}(A) \leq 2 \cdot \mathbf{Adv}_{Enc}^{\text{PRP}}(B) + \frac{q_e^2}{2^{\beta+1}},$$

where  $B$  is a PRP distinguisher with advantage  $\mathbf{Adv}_{Enc}^{\text{PRP}}$  for any time  $t' = t + O(q')$  and  $q'$  queries of total length  $\mu' = \mu_e$ , where  $q' \leq \rho + \left\lceil \frac{\mu_e}{\beta} \right\rceil$  ( $\rho$  is the total number of patterns found in  $q_e$ ).

For the authenticity proof of SLC, we extend the notion of unforgeability under chosen message attack [BN01] by adding the possibility of an adversary to authenticate parts of the message  $M$ . Otherwise, the scheme cannot be fault-tolerant in any sense.

First, we show that the proposed SSWC MAC is a secure MAC under the stated notion and its security is implied by the secure WC MAC ([Sho96] and [Ber05]). We also prove the suggested underlying primitive PolyHash for the SSWC scheme to be  $\varepsilon$ -AXU (similarly to [MV04]) hash function family.

The ability of an attacker to forge SSWC message parts under our authenticity notion does not allow active attacks like forging with swapped message parts. The fresh and non-repeating nonces for every new message prevent the adversary from mounting this type of attacks. An active adversary can only succeed in inserting, removing or altering

message parts, which the receiver would not be able to authenticate correctly. Still, the receiver would get an indication for the erroneous part and skip it for further processing the unaltered correctly placed parts.

The authenticity of the AE SLC scheme is formalized similarly to [MV04] as:

**Theorem 2 (Authenticity of SLC)** Let  $Enc$  be a family of permutations from  $\mathcal{G} = GF(2^\beta)$  to  $\mathcal{G} = GF(2^\beta)$ , and let the underlying MAC of the SLC scheme be  $\varepsilon'$  secure. Then for any time  $t$ ,  $q$  (authenticated encryption and decryption oracle queries), of total length  $\mu$ , an adversary  $A$  has forging advantage against SLC

$$\text{Adv}_{\text{SLC}[Enc]}^{\text{auth}}(A) \leq 2 \cdot \text{Adv}_{Enc}^{\text{prp}}(B) + \frac{q^2}{2^{bl} + 1} + \varepsilon',$$

where  $\mathcal{B}$  is a PRP distinguisher with advantage  $\text{Adv}_{Enc}^{\text{prp}}$  for any time  $t' = t + O(q')$ , and  $q'$  queries of total length  $\mu' = \mu$ , where  $q' \leq \rho + \left\lceil \frac{\mu}{\beta} \right\rceil$  ( $\rho$ , total number of patterns found in  $q$ ).

Finally, the security of the SLC scheme is concluded by known implications [BN01].

## 5 Conclusions

In this paper, we presented a new, provably secure, self-synchronizing authenticated encryption mode of operation, SLC, that can efficiently handle short packets. In view of current technology, this property has significant advantages, e.g., in rapidly emerging VoIP applications as there the IP packets are very small and the use of cryptographic functions in embedded systems (like phones) need to be light-weight and highly efficient. Subsequent work will focus on better bounds for the security proofs, on benchmarking and comparison with other AE schemes, and, on the integration in a full interoperable security framework supporting secure end-to-end communication in heterogeneous networks like [AS03] or SCIP/FNBDT [FNB03].

## References

- [AGPS01] A. Alkassar, A. Geraidy, B. Pfitzmann, and A.-R. Sadeghi. Optimized Self-Synchronizing Mode of Operation. In Matsui [Mat01], pages 87–91.
- [AS03] Ammar Alkassar and Christian Stübke. Security Framework for Integrated Networks. In *MILCOM IEEE Military Communications Conference 2003*, Boston, Massachusetts, October 2003.
- [BDJR97] M. Bellare, A. Desai, E. Jorjani, and P. Rogaway. A Concrete Security Treatment of Symmetric Encryption: Analysis of the DES Modes of Operation. In *38th Annual Symposium on Foundations of Computer Science*, pages 394–403. IEEE Computer Society, October 20–22, 1997.

- [Ber05] D. J. Bernstein. Stronger Security Bounds for Wegman-Carter-Shoup Authenticators. In Ronald Cramer, editor, *Advances in Cryptology — EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 164–180. Springer, May 22–26, 2005.
- [BN01] M. Bellare and C. Namprempe. Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm. In Tatsuaki Okamoto, editor, *Advances on Cryptology — ASIACRYPT 2000*, volume 1976 of *LNCS*, pages 531–545. Springer, December 3–7, 2001. ISBN 3-540-41404-5.
- [BRW04] M. Bellare, P. Rogaway, and D. Wagner. The EAX Mode of Operation. In Roy and Meier [RM04], pages 389–407.
- [CW79] L. L. Carter and M. N. Wegman. Universal Classes of Hash Functions. *jcss*, 18(2):143–154, April 1979.
- [FNB03] *FNBDT: End-to-End Security Workshop*, Royal Military Academy, Brussels, February 2003. NATO NC3A CIS.
- [GD01] V. D. Gligor and P. Donescu. Fast Encryption and Authentication: XCBC Encryption and XECB Authentication Modes. In Matsui [Mat01], pages 92–108.
- [Hey03] H. M. Heys. Analysis of the Statistical Cipher Feedback Mode of Block Ciphers. *IEEE Trans. Computers*, 52(1):77–92, January 2003.
- [JR99] O. Jung and C. Ruland. Encryption with Statistical Self-Synchronization in Synchronous Broadband Networks. In Ç. Koç and C. Paar, editors, *Cryptographic Hardware and Embedded Systems, First International Workshop, CHES'99*, volume 1717 of *LNCS*, pages 340–352. Springer, August 12–13, 1999.
- [Jut01] C. S. Jutla. Encryption Modes with Almost Free Message Integrity. In Birgit Pfitzmann, editor, *Advances in Cryptology — EUROCRYPT 2001*, volume 2045 of *LNCS*, pages 529–544. Springer, 6–10 May 2001.
- [Kra94] H. Krawczyk. LFSR-based Hashing and Authentication. In Y. Desmedt, editor, *Advances in Cryptology—CRYPTO '94*, volume 839 of *LNCS*, pages 129–139. Springer, August 21–25 1994.
- [KVV04] T. Kohno, J. Viega, and D. Whiting. CWC: A High-Performance Conventional Authenticated Encryption Mode. In Roy and Meier [RM04], pages 408–426.
- [Luc05] S. Lucks. Two-Pass Authenticated Encryption Faster than Generic Composition. In H. Gilbert and H. Handschuh, editors, *FSE 2005*, volume 3557 of *LNCS*, pages 284–298, Paris, France, February 21–23, 2005. Springer.
- [Mat01] M. Matsui, editor. *FSE 2001*, volume 2355 of *LNCS*, Yokohama, Japan, 2–4 April 2001. Springer, 2002.
- [MV04] D. A. McGrew and J. Viega. The Security and Performance of the Galois/Counter Mode (GCM) of Operation. In A. Canteaut and K. Viswanathan, editors, *INDOCRYPT 2004*, volume 3348 of *LNCS*, pages 343–355. Springer, December 20–22, 2004.
- [RBB03] P. Rogaway, M. Bellare, and J. Black. OCB: A Block-Cipher Mode of Operation for Efficient Authenticated Encryption. *ACM TISS*, 6(3):365–403, August 2003.
- [RM04] B. Roy and W. Meier, editors. *FSE 2004*, volume 3017 of *LNCS*, New Delhi, India, February 5–7, 2004. Springer.

- [Sho96] V. Shoup. On Fast and Provably Secure Message Authentication Based on Universal Hashing. In Neal Koblitz, editor, *Advances in Cryptology—CRYPTO '96*, volume 1109 of *LNCs*, pages 313–328. Springer, August 18–22 1996.
- [WC81] M. N. Wegman and L. Carter. New Hash Functions and Their Use in Authentication and Set Equality. *jcss*, 22(3):265–279, 1981.
- [WHF02] D. Whiting, R. Housley, and N. Ferguson. Counter with CBC-MAC (CCM). Technical report, Submission to NIST, AES modes of operations, June 2002.